z/OS V2.5 IBM Education Assistant

Solution Name: Miscellaneous functions for v2.5

Solution Element(s): z/OS Client Web Enablement Toolkit (HTTP/HTTPS enabler)

July 2021

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

- Describe the following enhancements for the HTTP/HTTPS enabler portion of the z/OS client web enablement toolkit. These enhancements are included in z/OS 2.5 release and were also rolled back to previous releases.
  - Enhancements
  - Server Name Indication
  - PATCH and OPTIONS HTTP Methods
  - Support for TLS 1.3 in the SSL path

Overview – Tracing Enhancements

- Who (Audience)
  - Users of the HTTP/HTTPS enabler portion of the toolkit

- What (Solution)
  - Each trace line is prefixed with date, time, pthread, and pid information
  - Existing trace option enhanced to redact information
  - System SSL tracing option
  - Support for runtime configuration of tracing options

- Wow (Benefit / Value, Need Addressed)
  - REXX Programmers who use HWTH_SSL_USE are now able to capture corresponding z/OS System SSL tracing when debugging their applications
  - Programmers can now more easily and effectively debug their applications and share their traces with L2 without fear or exposing sensitive or personal information

Usage & Invocation –Tracing Enhancements

The tracing enhancement consists of the following additions:

- Updates to the existing HWTH_OPT_VERBOSE connection option
  - **HWTH_VERBOSE_ON**
    - The behavior of the existing value is updated to no longer output any sensitive information
    - Each trace statement will contain the ISO 8601 date and time, pthread, pid, and ppid information
  - **HWTH_VERBOSE_UNREDACTED**
    - Brand new value is introduced to enable the user to see all the previous output except for Proxy-Authorization and Authorization header values
    - Each trace statement will contain the ISO 8601 date and time, pthread, pid, and ppid information
- A new connection option to enable SSL trace
  - **HWTH_OPT_SSLTRACE**
- Support for runtime configurations of
  - **HWTH_OPT_SSLTRACE**
  - **HWTH_OPT_VERBOSE**
  - **HWTH_OPT_VERBOSE_OUTPUT**

Usage & Invocation – HWTH_VERBOSE_ON

Behavior of HWTH_OPT_VERBOSE connection option value **HWTH_VERBOSE_ON** was updated to automatically redact information that might be considered sensitive:

- Redact ?query#frag from the request line

  Request line in trace before and after

```
GET /foo/bar?mysecret=here HTTP/1.1
GET /foo/bar?[redacted] HTTP/1.1
```

- Redact the first and last 40 bytes of the request and response body

  Response body in the trace before and after

```
t: Client received 1650 byte response:
t:
Response:
First 40 (of 1270) bytes: <!doctype html>.<html>.<head>.    <title (Hex: 4c5a849683a3a897854088a394936e154c88a394936e154c888581846e15404040404ca389a39385)
t: Last 40 (of 1270) bytes: ation...</a></p>.</div>.</body>.</html>. (Hex: 81a38996954b4b4b4c61816e4c61976e154c618489a56e154c61829684a86e154c6188a394936e15)

t: Client received 1650 byte response.
```

Usage & Invocation – HWTH_VERBOSE_ON continued

- Redact header value *unless:*
  - header value is associated with a ‘known’ header listed in RFC7231 will remain visible
    - z/OS MVS Programming: Callable-Services for High-Level Languages: HWTH_VERBOSE_ON description includes a complete list of these headers
      - » sample: Accept, Accept-Charset, Accept-Encoding

```
Request headers in trace before and after
        Proxy-Authorization: Basic dW5rdXNlcjp1bmtwYXNz
        Authorization: Basic dW5rdXNlcjp1bmtwYXNz

        Proxy-Authorization: [redacted]
        Authorization: [redacted]
```

  - header is a **Set-Cookie** in which case redact the value only and continue to show **Expires, Max-age, Domain, Path, Secure, HttpOnly** attribute values (av)

```
Example Set-Cookie Response Header
      Set-Cookie: 1P_JAR=2019-04-25-19; expires=Sat, 25-May-2019 19:24:27 GMT; path=/; domain=.google.com
      Set-Cookie: 1P_JAR=[redacted]; expires=Sat, 25-May-2019 19:24:27 GMT; path=/; domain=.google.com
```

cookie-pair

cookie-av’s

Usage & Invocation – HWTH_VERBOSE_UNREDACTED

A new value, **HWTH_VERBOSE_UNREDACTED**, was added for HWTH_OPT_VERBOSE connection option that provides unredacted tracing content

- Will output ?query#frag from the request line

```
GET /foo/bar?mysecret=here HTTP/1.1
```

- Will output all headers except for values associated with the following two authorization related headers:

```
Proxy-Authorization: [redacted]
Authorization: [redacted]
```

- Will output the first and last 40 bytes of the request and response body

Usage & Invocation – HWTH_OPT_VERBOSE output

The following enhancements were made for the tracing content generated when

HWTH_OPT_VERBOSE option is enabled (HWTH_VERBOSE_ON, HWTH_VERBOSE_UNREDACTED)

- trace lines are now prepended with the following information:
  - date and time in ISO8601 format, GMT
  - pthread
  - pid
  - ppid

- trace also provides details regarding the security type (SSL protocol), cipher specs, and SNI associated with the connection when using the System SSL path (HWTH_OPT_USE_SSL = HWTH_SSL_USE)

  t: Security type:<SSL protocol value> Cipher spec:<cipher value> SNI:<**on** if SNI is used|**off** if SNI not used>

Usage & Invocation – HWTH_OPT_VERBOSE output continued

Example prefix output, ISO 8601 format (GMT):

```
          1    1    2    2    3    3    4    4    5    5    6    6
....5....0....5....0....5....0....5....0....5....0....5....0....5
2019-07-10T01:57:24.105881Z 0BD2580000000000 0050331656 0016777222
```

Date and time in ISO8601 format, GMT

pthread

pid

ppid

```
  Menu  Utilities  Compilers  Help
 BROWSE    /SYSTEM/tmp/xzopsdd.$EXAMPLE.timeout1__:0001.07-18-2019_22:38:37.92              Line 0000000026 Col 001 125
2019-07-18T22:38:38.636676Z 0BB2480000000000 0067108899 0000000001 t-Entry: restoreSignal
2019-07-18T22:38:38.636678Z 0BB2480000000000 0067108899 0000000001 t: restoring signal: SIGPIPE
2019-07-18T22:38:38.636681Z 0BB2480000000000 0067108899 0000000001 t-Exit: restoreSignal
2019-07-18T22:38:38.636686Z 0BB2480000000000 0067108899 0000000001 t: No applicable peerid.
2019-07-18T22:38:38.636688Z 0BB2480000000000 0067108899 0000000001 t-Entry: ignoreSignal
2019-07-18T22:38:38.636691Z 0BB2480000000000 0067108899 0000000001 t: now ignoring signal: SIGPIPE
2019-07-18T22:38:38.636693Z 0BB2480000000000 0067108899 0000000001 t-Exit: ignoreSignal
2019-07-18T22:38:38.636695Z 0BB2480000000000 0067108899 0000000001 t: Invoke gsk_secure_socket_init()
2019-07-18T22:38:38.707636Z 0BB2480000000000 0067108899 0000000001 t-Entry: restoreSignal
2019-07-18T22:38:38.707641Z 0BB2480000000000 0067108899 0000000001 t: restoring signal: SIGPIPE
2019-07-18T22:38:38.707644Z 0BB2480000000000 0067108899 0000000001 t-Exit: restoreSignal
2019-07-18T22:38:38.707648Z 0BB2480000000000 0067108899 0000000001 t: A full handshake was required.
2019-07-18T22:38:38.707649Z 0BB2480000000000 0067108899 0000000001 t-Exit: initSSLEnv
2019-07-18T22:38:38.707651Z 0BB2480000000000 0067108899 0000000001 t: Socket was secured with toolkit options: 1
2019-07-18T22:38:38.707657Z 0BB2480000000000 0067108899 0000000001 t: Security type:'TLSV1.2' Cipher spec:'0035' SNI:off
2019-07-18T22:38:38.707659Z 0BB2480000000000 0067108899 0000000001 t-Exit: iconnImpl
2019-07-18T22:38:38.732946Z 0BB2480000000000 0067108899 0000000001 t-Entry: sendrqst
2019-07-18T22:38:38.732998Z 0BB2480000000000 0067108899 0000000001 t-Entry: sendrqstImpl
2019-07-18T22:38:38.733001Z 0BB2480000000000 0067108899 0000000001 t-Entry: appendRequestLine
2019-07-18T22:38:38.733003Z 0BB2480000000000 0067108899 0000000001 t: No proxy is being used for the request
2019-07-18T22:38:38.733007Z 0BB2480000000000 0067108899 0000000001 [HWTHCKST] (no request cookies specified)
2019-07-18T22:38:38.733010Z 0BB2480000000000 0067108899 0000000001 [HWTHCKST] getCookieHeader() - No applicable cookies found
2019-07-18T22:38:38.733011Z 0BB2480000000000 0067108899 0000000001 t: No applicable cookies found
2019-07-18T22:38:38.733015Z 0BB2480000000000 0067108899 0000000001 t: * * * * * HTTP REQUEST HEADERS * * * * *
2019-07-18T22:38:38.733017Z 0BB2480000000000 0067108899 0000000001 t: GET /ip HTTP/1.1.
Host: barney.rtp.raleigh.ibm.com:51002.
 Command ===>                                                                  Scroll ===> PAGE
```

Usage & Invocation – HWTH_OPT_SSLTRACE

A new connection option is added that allows applications to programmatically enable SSL tracing when using the System SSL path (HWTH_OPT_USE_SSL = HWTH_SSL_USE)

HWTH_OPT_SSLTRACE

- This option is disabled by default
- To enable the user must provide a fully qualified z/OS UNIX file system location to where the System SSL trace should be directed
- When enabled:
  - GSK_TRACE is set to 255 (0xFF)
  - GSK_TRACE_FILE is set to the user specified location
  - i.e. /tmp/gsktrace.%.trc where % will be replaced with the process id of the encompassing process issuing the REST API
  - The user can run the **gsktrace** command on the produced file to create a readable copy of the SSL trace information

GSK TRACE reference: https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.4.0/com.ibm.zos.v2r4.gska100/sssl2dia1023934.htm

Usage & Invocation – External Tracing Configuration

In addition to the other features, support was added that allows the usage of the following connection options as runtime environment variables:

- HWTH_OPT_SSLTRACE
- HWTH_OPT_VERBOSE
- HWTH_OPT_VERBOSE_OUTPUT

The value of these runtime environment variables will take precedence over any value specified for that variable via the HWTHSET service. For example, if application **`hwthxrx1`** sets verbose to off:

```
address hwthttp  "hwthset ReturnCode SessionHandle1 HWTH_OPT_VERBOSE HWTH_VERBOSE_OFF DiagArea."
```

The user running the application, can set **HWTH_OPT_VERBOSE** environment variable to **HWTH_VERBOSE_ON** to override the application setting and enable tracing.

NOTE: This feature is **not** applicable for System REXX environments

Usage & Invocation – External Tracing Configuration examples

If the application is running in z/OS UNIX, an **Language Environment® (LE) POSIX(ON) environment**, the user can set the override environment variables using the `export` command

```
export HWTH_OPT_VERBOSE=HWTH_VERBOSE_UNREDACTED
export HWTH_OPT_SSLTRACE=/user/hwth/gskssl.trc
```

If application is NOT running in an **LE POSIX(ON) environment,** the user can set the override environment variables using the `CEEOPTS DD` statement.

- https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.4.0/com.ibm.zos.v2r4.ceea200/ceedd.htm

- Batch example using instream JCL to set runtime environment variables for rexx exec `GGEXP06`:

```
EDIT       HWT.GORELIK.UNITTEST.JCL(GGEXP06) - 01.10            Columns 00001 00080
Command ===>                                                    Scroll ===> PAGE
****** ********************************* Top of Data ***********************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG>           your edit profile using the command RECOVERY ON.
000001 //GGEXP06  JOB CLASS=J,NOTIFY=&SYSUID,MSGLEVEL=1,MSGCLASS=A,REGION=0K
000002 //*-------------------------------------------------------------
000003 //* MGM:  CHANGED TO USE IKJEFT1A TO PROGAGATE ERROR FROM EXEC
000004 //*
000005 //* USING IKJEFT1A TO SUBMIT THE EXEC IN THE TSO ENVIRONMENT
000006 //*-------------------------------------------------------------
000007 //TMP     EXEC PGM=IKJEFT1A,DYNAMNBR=30,REGION=0M,TIME=9999
000008 //SYSEXEC  DD  DSN=HWT.GORELIK.REXX,DISP=SHR
000009 //SYSTSPRT DD  SYSOUT=A
000010 //*
000011 //GGDD     DD  PATH='/tmp/gorelik/gorelik.trace1'
000012 //*
000013 //CEEOPTS  DD *
000014   ENVAR("HWTH_OPT_VERBOSE=HWTH_VERBOSE_UNREDACTED",
000015   "HWTH_OPT_VERBOSE_OUTPUT=GGDD",
000016   "HWTH_OPT_SSLTRACE=/tmp/gorelik/gskssl.trc")
000017 /*
000018 //SYSTSIN  DD  *
000019  %GGEXP06
000020 /*
****** ********************************* Bottom of Data ********************************
```

Usage & Invocation – External Tracing Configuration examples

Additional example using the CEEOPTS DD statement:

- TSO/E example referencing a sequential data set `HWT.GORELIK.TRACING.CONFIG` that contains the runtime environment variables

```
EDIT       HWT.GORELIK.TRACING.CONFIG                          Data set saved
Command ===> ____________________________________________ Scroll ===> PAGE
****** ********************************* Top of Data ***********************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001  ENVAR("HWTH_OPT_VERBOSE=HWTH_VERBOSE_UNREDACTED",
000002  "HWTH_OPT_VERBOSE_OUTPUT=GGDD",
000003  "HWTH_OPT_SSLTRACE=/tmp/gorelik/ggsksslc.trc")
****** ******************************** Bottom of Data ********************************
```

```
 READY
alloc ddname(CEEOPTS) da('hwt.gorelik.tracing.config') shr
 READY
alloc ddname(GGDD) path('/tmp/gorelik/gorelik.trace')
 READY
ex 'hwt.gorelik.rexx(ggexp06)'
```

- REXX applications running in z/OS UNIX example where a wrapper rexx exec is used to allocate the required ddnames and then calls the rexx application that should be impacted by the runtime environment variables

```
000001 /* REXX */
000002 call bpxwdyn "alloc dd(GGDD) path('/tmp/gorelik/gorelik.trace3')"
000003 call bpxwdyn "alloc dd(CEEOPTS) da('HWT.GORELIK.TRACING.CONFIG') shr"
000004 call './ggexp06'
```

Overview - Server Name Indication (SNI)

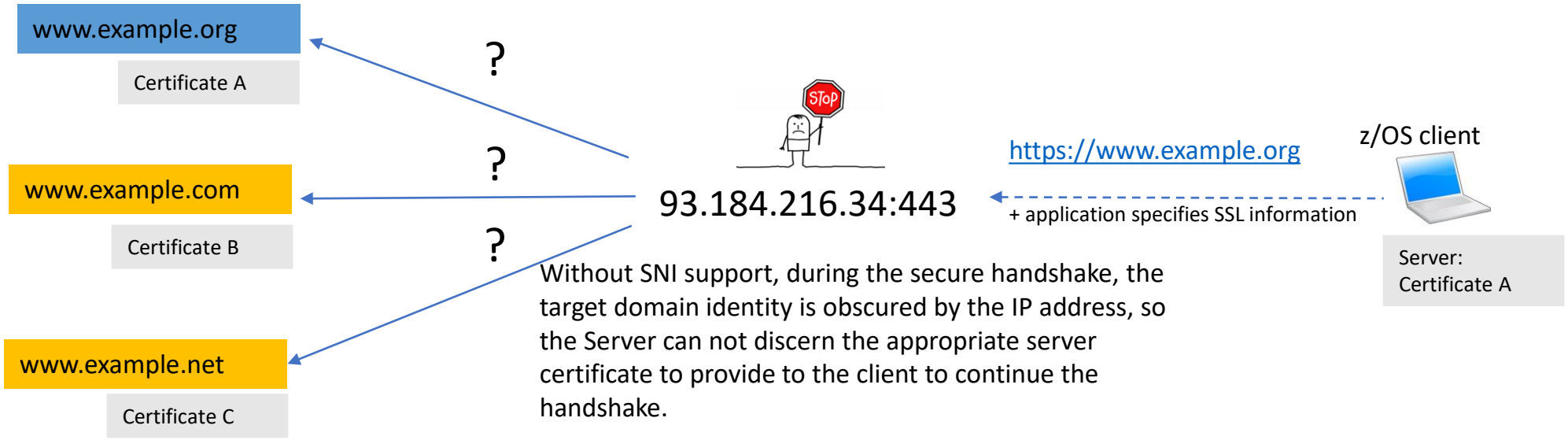
- Who (Audience)
  - HTTP/HTTPS enabler users that explicitly specify SSL/TLS security configuration and use the URI domain name system (DNS) format
- What (Solution)
  - Enhanced HWTH_SSL_USE option to automatically include an SNI extension when the connection URI is in the DSN format for a secure request
- Wow (Benefit / Value, Need Addressed)
  - The client can now reliably reach and interact with a specific target domain on a multi domain server.

Usage & Invocation – SNI

Background for Server Name Indication (SNI)

IP address: 93.184.216.34
Domain Name System (DNS): www.example.org

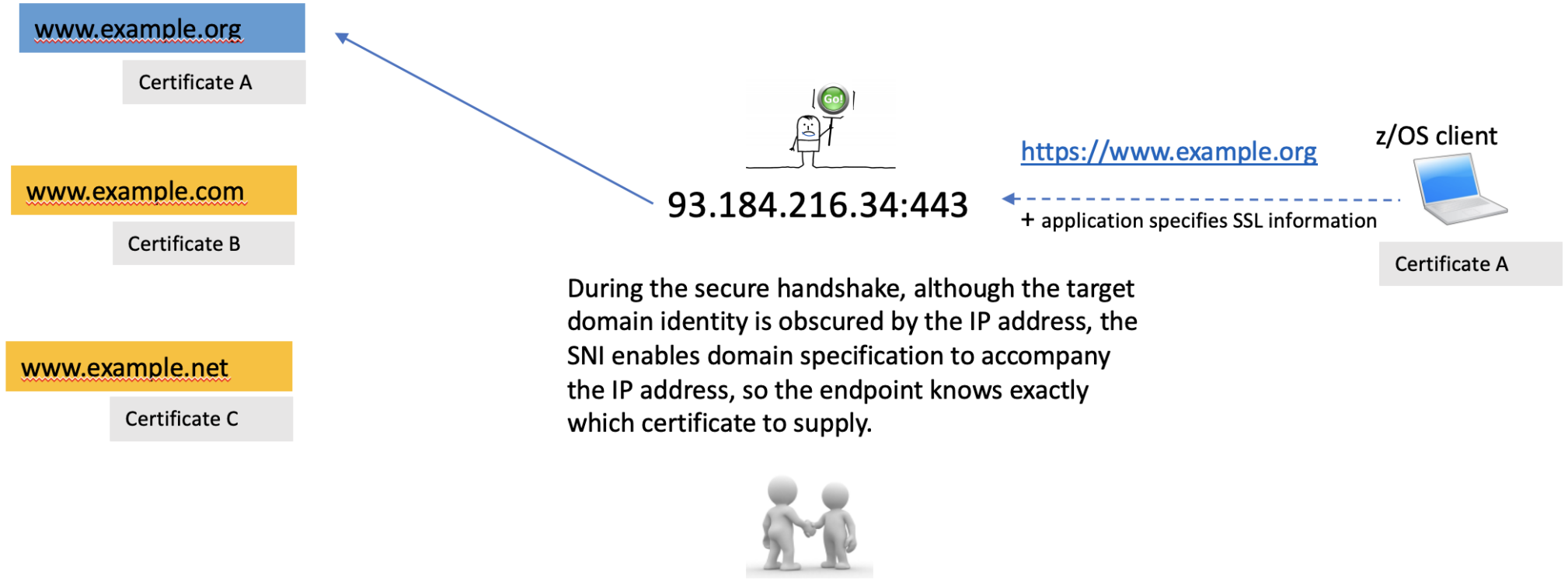
- Applicable for secure connections, HTTPS
- Allows for a single IP address to support multiple domains
  - Security is based on the domain name instead of the IP address
    - Each domain can have its own unique SSL certificate
    - *Alternatively, and more expensive: dedicated IP address per domain*



Without SNI support, during the secure handshake, the target domain identity is obscured by the IP address, so the Server can not discern the appropriate server certificate to provide to the client to continue the handshake.

Usage & Invocation – SNI

The **HWTH_SSL_USE** option was enhanced to automatically include an SNI extension when the connection URI is in the DNS format for a secure request [HTTPS]



Overview – PATCH and OPTIONS

- Who (Audience)
  - Toolkit HTTP/HTTPS enabler users that want to issue requests that use either the PATCH HTTP Method or the OPTIONS HTTP Method
- What (Solution)
  - Support for PATCH and OPTIONS HTTP Methods
- Wow (Benefit / Value, Need Addressed)
  - Programmers can now incorporate usage of requests that required PATCH and OPTIONS HTTP Method

Usage & Invocation – PATCH and OPTIONS

New values were added for the existing HWTH_OPT_REQUESTMETHOD option

- **HWTH_HTTP_REQUEST_PATCH**
  - Use the PATCH method
  - Used to do partial resource modification, compared to the HTTP PUT method that focuses on a complete replacement of a document. [RFC 5789]

- **HWTH_HTTP_REQUEST_OPTIONS**
  - Use the OPTIONS method
  - Returns headers, which will advertise the server's abilities, and a possible HWTH_OPT_RESPONSEBODY. [RFC 7231]

Overview – TLS 1.3

- Who (Audience)
  - Toolkit HTTP/HTTPS enabler users that explicitly specify the SSL/TLS security configuration details (HWTH_SSL_USE)
- What (Solution)
  - Support for TLS version 1.3
- Wow (Benefit / Value, Need Addressed)
  - Toolkit applications can now take advantage of the latest encryption protocol

Usage & Invocation – TLS 1.3

New value was added for the existing **HWTH_OPT_SSLVERSION** connection option

- **HWTH_SSLVERSION_TLSv13**
  - Support for TLS version 1.3
  - Requires ICSF

Interactions & Dependencies

- Software Dependencies
  - TLS 1.3 - Requires ICSF
- Hardware Dependencies
  - None
- Exploiters
  - None

Upgrade & Coexistence Considerations

- To exploit this solution, all systems in the Sysplex must be at the new z/OS level: No

Installation & Configuration

- APAR OA58707
  - includes:
    - Tracing Enhancements
    - Server Name Indication
    - and OPTIONS HTTP Methods
  - Rolled down to z/OS 2.3
  - IPL is required after applying the APAR

- APAR OA58708
  - includes:
    - Support for TLS 1.3 in the SSL path
  - Rolled down to z/OS 2.4
  - IPL is required after applying the APAR

Summary

z/OS 2.5 release of z/OS client web enablement toolkit includes a variety of enhancements for the HTTP/HTTPS protocol enabler

- Programmers can more easily debug their applications by taking advantage of the additional tracing features
- Applications can now interact with servers that use the same IP address for multiple domains
- Applications can take advantage of requests that use either the PATCH or OPTIONS HTTP Methods
- Applications can take advantage of the latest SSL/TLS protocol: TLS 1.3

Appendix

Publications

- **z/OS MVS Programming: Callable Services for High-Level Languages**
  - Complete z/OS client web enablement toolkit documentation
- **z/OS MVS System Messages, Volume 6 (GOS – IEA)**
  - Toolkit message documentation
- **z/OS MVS System Codes**
  - Toolkit abend ’04D’x documentation

Additional samples for z/OS client web enablement toolkit are provided via github:

- https://github.com/IBM/zOS-Client-Web-Enablement-Toolkit