CICLIENT Programmer Documentation

April 11, 2023

Introduction

This document describes a component named CICLIENT developed by Charles Mills Consulting (CMC) for NewEra. CICLIENT is intended to be called from NewEra ICEdirect Certificate Intelligence ("CI"). The intended audience for this document is an experienced Rexx language developer. CMC provides no end-user documentation for CICLIENT.

All trademarks used in this document are the exclusive property of their respective owners. No association with CMC is implied.

CICLIENT is a z/OS program written primarily in IBM XLC C++. CICLIENT could be considered as an interface between CI and IBM z/OS System SSL. (System SSL is the primary subsystem by which z/OS components, including PAGENT AT-TLS, implement TLS.)

https://www.ibm.com/docs/en/zos/2.5.0?topic=services-zos-cryptographic-system-ssl-programming

CICLIENT extends Certificate Intelligence in two ways: (1) by diagnosing the certificate response of a *remote* Linux, Windows or z/OS server (as well as a local z/OS server), while existing CI is limited to the *local LPAR* certificate databases; and (2) while existing CI is essentially "static" ("here is what exists in the certificate databases") CI with CICLIENT is "dynamic" ("we tried actually using the certificates and here is what happened").

CICLIENT supports certificate databases implemented by a security subsystem (IBM RACF or Broadcom ACF2 or Top Secret), possibly in conjunction with ICSF; or databases implemented by IBM gskkyman.

CICLIENT accepts parameters from CI, attempts to connect to a specified server as a client using System SSL, and reports the results, including certificates involved (see certificate detail below). Parameters and results (other than a standard return code) are in a character form intended to facilitate usage with Rexx. Note that this "loose" character-based interface facilitates de-coupled development, in which CICLIENT may be exercised by CMC with a "quick and dirty" Rexx program without significant involvement of NewEra developers.

This description will be subject to change by agreement between NewEra and CMC as CICLIENT is developed.

Usage Scenarios

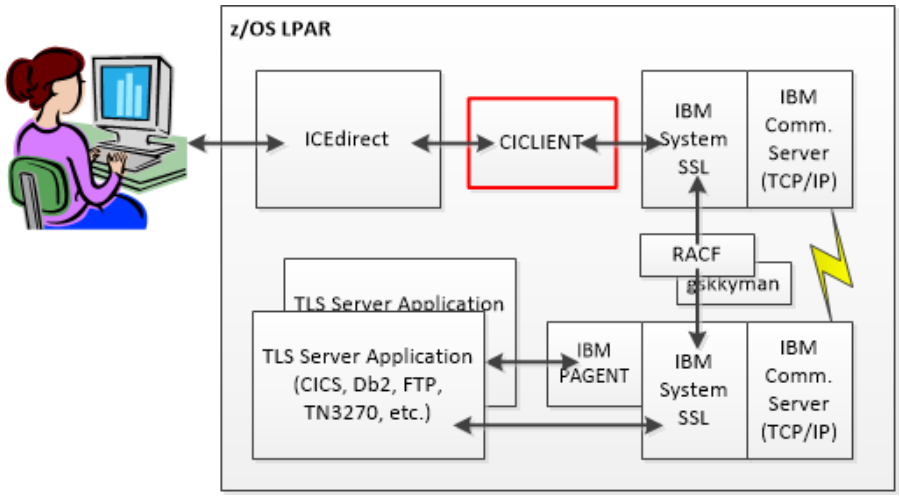
Scenario 1: The customer is configuring a new server (CICS, Db2, FTP, TN3270, WebSphere, etc.) on their z/OS LPAR and wish to confirm its correct operation. See Configuration 1 below.

Scenario 2: The customer wishes to validate or troubleshoot the connection between z/OS and some remote server (Web, FTP, etc.). See Configuration 2 below.

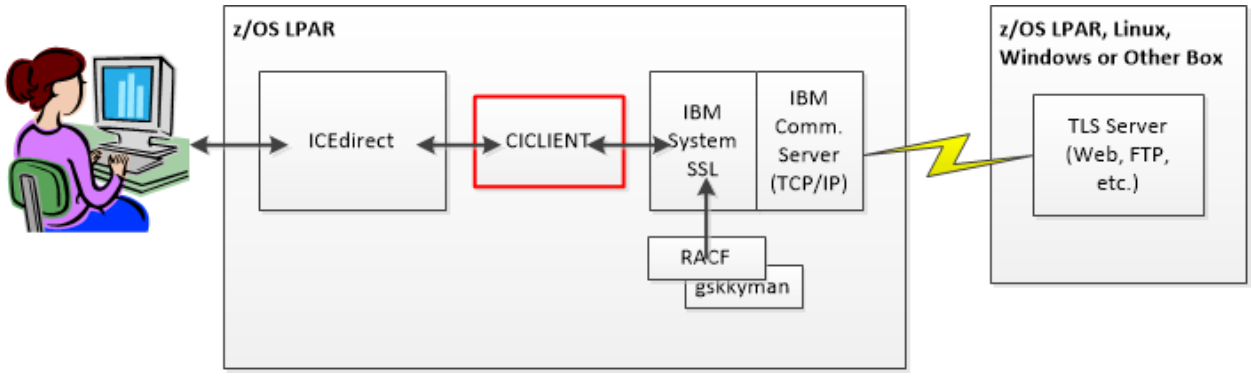
Scenario 3: The customer wishes to perform "what-if" analysis such as "what will the impact be if we disable the use of TLSv1.1?" without risk to a production configuration. See either configuration below.

Additional scenarios involving validation, troubleshooting or "what-if" analysis are possible using either of the configurations below.

Configuration 1: Validating or Diagnosing the Configuration of or Connection to a Local Server



Configuration 2: Validating or Diagnosing the Connection to a Remote Server
(Note that from CICLIENT's point of view, this configuration is indistinguishable from Configuration 1.)



Input Parameters

The CI input to CICLIENT is a single character string passed as argument one of a z/OS Rexx function call. The individual CICLIENT parameters are separated in the argument string by one or more blanks. (Note that the well-known 100-character parameter limit applies only to JCL jobsteps and TSO commands and is not relevant to Rexx functions.)

The input parameter format is modeled on BPXWDYN.
https://www.ibm.com/docs/en/zos/2.3.0?topic=output-keywords

Input parameters are in one of two formats:

> Keyword, for example, FIPS
>
> Keyword plus argument(s) separated by commas plus one or more blanks following the comm, for example PORT(443) or PROTOCOL(NOSSLV3, NOTLSV1.1, NOTLSV1.2)

Any argument *may* be quoted; any argument that contains a blank or comma *must* be quoted. Single quotes (apostrophes) or double quotes (standard quotation marks) may be used. As with BPXWDYN, doubled interior quotes are not supported. Relaxing the BPXWDYN restriction, a quote character within quotes is permitted so long as it is not immediately followed by a comma, blank or a right parenthesis: `"Paul's Cert"` and `'Paul's Cert'` are correct; ~~`'Paul''s Cert'`~~ is not (unless, of course, the value actually contains two apostrophes).

Parameters are processed from left to right, with subsequent keywords and arguments overriding earlier keywords and arguments. Keywords and arguments are case-independent: FIPS, fips and Fips are all logically equivalent. Arguments are treated as upper case.

IPADDR and URL

The IPADDR and URL keywords provide alternative ways of specifying the IP address of the server, and in addition, the URL keyword provides a name for validating the received server certificate.

- If you specify URL and not IPADDR then CICLIENT resolves the URL to an IP address and uses it for the session. If the URL fails to validate the certificate it is a non-fatal error.
- If you specify IPADDR and not URL then CICLIENT uses the specified IP address for the session, and issues a warning message that it could not validate the certificate name.
- If you specify both then CICLIENT uses address in IPADDR for the session. If the URL fails to validate the certificate it is a non-fatal error.

Ciphers and Similar Arguments

The argument for keywords that specify cipher suites, algorithm pairs, or similar 4-character values may be specified as a single argument comprising all of the desired values `(1301130213030005)` or as multiple arguments each consisting of one or more 4-character values separated by commas plus zero or more optional blanks `(1301,1302,1303,0005)` or `(13011302, 1303,  0005)`. All of the forms are equivalent.

The following parameters are accepted:

Keyword	Format	Description	Default
CERTSIGALGS	Keyword plus argument	Specifies hash and signature algorithm pair specifications that are supported by the client or server as a string consisting of one or more 4-character values in order of preference for use in digital signatures of X.509 certificates. If specified, CERTSIGALGS overrides SIGALGS with respect to certificate signatures. See https://www.ibm.com/docs/en/zos/2.4.0?topic=programming-cipher-suite-definitions#csdcwh__sign2and3 for a list of valid values.	No overriding signature pairs
CIPHERS	Keyword plus argument. See Ciphers and Similar Arguments above	Specifies the ciphers to be used for the connection as one or more 4-digit values. See https://www.ibm.com/docs/en/zos/2.5.0?topic=programming-cipher-suite-definitions#csdcwh__tthcsd.	0005000400350036 003700380039002F 0030003100320033 000A001600130010 000D000900150012 000F000C00030006 000200010000
ECURVES	Keyword plus argument. See Ciphers and Similar Arguments above	Specifies the elliptic curves or groups that are supported by the client as one or more 4-character decimal values in order of preference for use. For TLS V1.0, TLS V1.1, and TLS V1.2 protocols, this list is used by the client to guide the server to the elliptic curves that are preferred when using ECC-based cipher suites. For the TLS V1.3 protocol, this list is used by the client to guide the server to the elliptic curves that are preferred and guide group selection to encrypt and decrypt TLS V1.3 handshake messages. The valid elliptic curves are 0019, 0021, 0023, 0024, 0025, 0029 and 0030. Only the last five are valid for TLSV1.3.	0021002300240025 0029 If TLSV1.3 is enabled, the effective default is 0029003000210023 00240025
EXPIREWARN	Keyword plus argument	Specifies the number of days for a certificate expiration warning threshold. If any processed certificate will expire within the specified number of days then a warning message is logged. Specify a value between 1 and 180 days, or to completely disable expiration warnings, specify 0.	7 days
EXPLICIT	Keyword plus argument	Specify EXPLICIT(FTP) to indicate that CICLIENT is to use the FTP convention of sending “AUTH TLS” after establishing a TCP/IP connection and before negotiating TLS.	Implicit TLS

Keyword	Format	Description	Default
FIPS	Keyword	Specifies FIPS mode. FIPS mode is a more restrictive TLS mode that conforms to NIST standard FIPS 140-2. FIPS mode requires a FIPS-compliant certificate database.	Not FIPS
IPADDR	Keyword plus Argument	Specifies the remote IP address to connect to, either in IPV4 format (192.168.1.6) or if IPV6 is specified, in IPV6 format (fe80::deaa:6525:f296:9a94). See IPADDR and URL above.	
IPV6	Keyword	Specifies that CICLIENT is to connect using IPv6.	CICLIENT uses IPv4
KEYRING	Keyword plus argument	Specifies the name of the ESM keyring or gskkyman database, in the conventional format, either [userid]/ringname or /complete/gsk/database/path.	*AUTH*/*, the CERTAUTH virtual key ring
KEYSHARES	Keyword plus argument. See Ciphers and Similar Arguments above	Specify the Key Share group definitions for TLSV1.3 as one or more 4-digit values. *Do not specify more Key Shares than ECURVES.*	002300240025
LABEL	Keyword plus argument	Specifies the client certificate label.	Keyring default certificate
OA64071	Keyword	Specifies that the PTF for APAR OA64071 has not been applied, and CICLIENT should use alternative certificate display logic. This keyword has no effect on CICILIENT running under z/OS V2R4 and below.	For z/OS V2R5 and above, CICLIENT assumes that the patch for APAR OA64071 has been applied.
PORT	Keyword plus argument	Remote TCP/IP port number from 1 to 65535.	443
PROTOCOL	Keyword plus arguments	Specifies the enabled protocols. SSL V2 is deprecated and is always disabled. Specify the enabled and disabled protocols, for example PROTOCOL(NOSSLV3,TLSV1.1,TLSV1.2)	Protocol Default SSLV3 NOSSLV3 TLSV1 NOTLSV1 TLSV1.1 TLSV1.1 TLSV1.2 TLSV1.2 TLSV1.3 TLSV1.3

Keyword	Format	Description	Default
SIGALGS	Keyword plus argument. See Ciphers and Similar Arguments above	Specifies the list of hash and signature algorithm pair specifications that are supported by the client as one or more 4-character values in order of preference for use in digital signatures of X.509 certificates and TLS handshake messages. See https://www.ibm.com/docs/en/zos/2.4.0?topic=programming-cipher-suite-definitions#csdcwh__sign2and3 for a list of valid values. The signature algorithm pair specification only has relevance for sessions using TLS V1.2 or higher protocols. See also CERTSIGALGS.	0601060305010503 0401040304020301 0303030202010203 0202 If TLSV1.3 is enabled, the effective default is 0804080508060601 0603050105030401 0403040203010303 0302020102030202
STACK	Keyword plus argument	Causes CICLIENT to use the specified TCP/IP stack name.	CICLIENT uses the default TCP/IP stack
TIMEOUT	Keyword plus argument	Specifies a connect timeout in seconds. Specify a value between 10 and 300 inclusive.	30
TRACE	Keyword plus optional arguments	Specifies that CICLIENT is to invoke the System SSL trace, format it, and return the formatted trace in the CICLIENT_TRACE. stem. You may optionally specify two arguments, *maxlines* and *level*. *Maxlines* specifies the maximum number of lines of trace data to be returned; *level* is a decimal number between 1 and 63 specifying the sum of the values for the types of events to be traced. The values are 1 -Trace function entry 2 - Trace function exit 4 - Trace errors 8 - Include informational messages 16 - Include EBCDIC data dumps 32 - Include ASCII data dumps The values may be specified or omitted, and if specified may be specified in either order. Values of 255 or less are assumed to specify *level*; values greater than 255 are assumed to specify *maxlines*.	5000,15
URL	Keyword plus argument	Required. Specifies the remote URL to connect to. See IPADDR and URL above.	None. URL is required.

Keyword	Format	Description	Default
VALMODE	Keyword plus argument	Specifies the certificate validation mode. RFC 2459, RFC 3280, and RFC 5280 describe differing methods of certificate validation. Specify ANY, RFC2459, RFC3280 or RFC5280.	ANY
VERBOSE	Keyword	Specifies that CICLIENT is to produce additional diagnostic messages. See Status Log below.	No additional diagnostic messages.

See Sample Invocation below for an example of a valid parameter string:

Output

The output of CICLIENT is a set of Rexx compound variables and a standard Rexx function return value.

The compound variables contain textual messages intended for direct user display by CI. CICLIENT follows the typical Rexx convention in which *stem*.0 contains the count of messages and *stem*.1, *stem*.2, etc. contain the actual messages.

Status Log

The log of CICLIENT activity is returned in a compound variable with a stem of CICLIENT_LOG. It consists of status, informational, diagnostic, warning and error messages.

Each message in CICLIENT_LOG begins with a one-character description code followed by a blank. It is intended that NewEra will choose whether to display the code or omit it from the display and instead use it to determine display characteristics (color, bold font, etc. – or omitted entirely). The codes will be as follows:

- I Informational message for the customer
- D Detailed status or diagnostic message, including Verbose output
- G Suggestion message; suggested user remediation for reported errors
- W Warning message
- E Error in protocol, certificate or cipher or similar
- S Invocation parameter or “should not occur” error from System SSL or z/OS
- C Critical error; CICLIENT abnormally terminated

It is CMC’s intention to make all messages as informative as possible. At a minimum, all error messages include the exact name of the failing function, any parameters necessary to pinpoint a specific invocation of that function, and a gsk_strerror() textual description of any error code. Additional information will be returned as agreed upon by NewEra and CMC.

See https://www.ibm.com/docs/en/zos/2.5.0?topic=reference-gsk-secure-socket-init Results for examples of the level of error detail that is returned. CICLIENT expands on the IBM error

message where possible and appropriate: “Make sure of blah-blah-blah. Sometimes this error is caused by X or by Y.”

In addition the status log is written to a single data set, *userid*.NEWERA.CICLIENT.LOG. This hard data set allows for debugging in those situations where an abnormal termination of CICLIENT precludes the setting of the LOG stem variable. (The data set is re-used and overwritten on each invocation.)

Certificate Information

Certificate information is returned in a set of compound variables named CICLIENT_CERT.*symbol*.*n*. Each variable contains one line of certificate information intended for direct display by CI. Symbol is obtained for the certificate index described immediately below.

The certificate information compound variables are indexed in a compound variable with a stem of CICLIENT_CERTINDEX. Each variable CICLIENT_CERTINDEX.*n* consists of a symbol name followed by one or more blanks and the name of a certificate. For example, CICLIENT_CERTINDEX.1 might contain

```
CHAIN1    GeoTrust TLS DV RSA Mixed SHA256 2020 CA-1
```

which would indicate that information for the certificate named GeoTrust TLS DV RSA Mixed SHA256 2020 CA-1 would be found in the compound variables named CICLIENT_CERT.CHAIN1.*n*. It is intended that CI would display the list of certificate names, and in response to the user’s clicking on one of the names, display the information in the variables.

Under all versions of z/OS, server certificate information from a successful negotiation is returned in stem variables named CICLIENT_CERT.SERVER.*n*. The information returned is as selected by NewEra from the *character format* “elements” of https://www.ibm.com/docs/en/zos/2.5.0?topic=reference-gsk-attribute-get-cert-info (elements with DER in their identifiers are binary). Each named element is returned in a separate stem variable, identified by a textual label indicating the specific element, and ready to be displayed by CI. CICLIENT_CERT.SERVER will be indexed in CICLIENT_CERTINDEX; it is not intended that CI hard code the symbol SERVER.

Under versions of z/OS from V2R5 forward, all certificates in the chain, whether the negotiation is successful or unsuccessful, are returned to CI in one or more sets of stem variables identified in CICLIENT_CERTINDEX.*n*. The information returned is in a format similar to the Server certificate information and includes the serial number and Subject and Issuer DNs as described in https://www.ibm.com/docs/en/zos/2.5.0?topic=reference-gsk-name-dn.

The following Rexx code is intended to show the relationship among the various certificate stems. (It is not intended for direct inclusion in CI.) It would print the certificate information for all available certificates.

```
DO i = 1 TO CICLIENT_CERTINDEX.0
```

```
PARSE VAR CICLIENT_CERTINDEX.i symbol name
SAY "Cert info for" name
DO c = 1 TO CICLIENT_CERT.symbol.0
  SAY CICLIENT_CERT.symbol.c
  END c
END i
```

Trace Data

If TRACE is specified in the parameters then trace data is returned in compound variables with a stem of CICLIENT_TRACE. The returned trace data looks like

```
11/17/2022-12:02:46 Thd-1 INFO cms_validate_certificate_mode_int(): ...
11/17/2022-12:02:46 Thd-1 ENTRY check_cert_extensions_3280_and_later(): ...
11/17/2022-12:02:46 Thd-1 ENTRY gsk_decode_certificate_extension(): ...
etc.
```

Rexx Function Result

CICLIENT returns a standard Rexx function result as follows:

0	Complete success; only I, D and G messages logged
4	Success with one or more W messages logged
8	E error message logged
12	S error message logged
16	C error message logged

Return Code

CICLIENT always returns a return code of 0, except in the case of an unhandled abnormal termination (ABEND). (Non-zero return codes from external functions generally cause the calling Rexx routine to terminate.)

Sample Invocation

```
PARM = "URL(WWW.NEWERA.COM) KEYRING(CMILL1/MYRING)"
RET = CICLIENT(PARM)
IF RET <> 0 THEN ...
```

Sample Program

A sample calling program written in z/OS Rexx is included as part of the CICLIENT deliverables.

Appendix

Sample Demonstration URLs

URL	Port	CA Root	In kyman_primary	In *AUTH*/*
delivery01-bld.dhe.ibm.com	21	DigiCert Global Root G2	Yes	
localhost	8201	Go Daddy Secure Certificate Authority – G2		V2R4
stackexchange.com	443	ISRG Root X1		
www.amazon.com	443	DigiCert Global Root CA	Yes	
cloudcompiling.com	443	Go Daddy Secure Certificate Authority - G2		V2R4
www.google.com	443	GlobalSign Root CA		
www.ibm.com	443	DigiCert Global Root CA	Yes	
www.microsoft.com	443	DigiCert Global Root G2	Yes	
www.newera.com	443	DigiCert Global Root CA	Yes	
www.newera-info.com	443	Baltimore CyberTrust Root		

Note: URL(localhost) accesses myicedirect, but the certificate validation will fail on a name mis-match. URL(delivery01-bld.dhe.ibm.com) requires EXPLICIT(FTP)

Gskkyman Key Databases

Note: The password for all gskkyman key databases is “password”. These key databases should not be used to store any private keys that have any security implications.

Name	Description
/u/cmill1/gsk/kyman_primary.kdb	Primary test key database. Intended to contain most common demonstration CA root certificates.
/u/cmill1/gsk/kyman_bad.kdb	Test of “bad” cases. Contains a purported CA root for Go Daddy Secure Certificate Authority - G2 and DigiCert Global Root CA