z/OS V2.5 IBM Education Assistant

Solution Name: Certificate failure diagnostics and extended master secret support

Solution Element(s): System SSL

July 2021

Agenda

- Trademarks
- Objectives
- Certificate failure diagnostics
  - Overview
  - Usage & Invocation
  - Interactions & Dependencies
  - Upgrade & Coexistence Considerations
  - Installation & Configuration
- Extended Master Secret (EMS)
  - 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

- At the end of this presentation, you will understand the following enhancements from System SSL:
  - Certificate failure diagnostics
    - An overview of the new certificate diagnostics features
    - How to enable and collect certificate diagnostics
    - What sort of failures and errors can the certificate diagnostics capture
    - What data is collected and provided by the diagnostics
  - Extended master secret support provided by System SSL
    - An overview of extended master secret support
    - What benefits the extended master secret support provides
  - Understand how these enhancements affect installation, migration and coexistence

Certificate Failure Diagnostics

Overview

- Who (Audience)
  - System Programmers that want diagnostic aids that help identify and resolve certificate validation failures that occur within System SSL
- What (Solution)
  - Enhance the certificate validation and handshake processes to collect diagnostic information during execution
  - Implement mechanisms to provide this diagnostic information to exploiters
    - New callback routine
    - Update **gsk_validate_certificate_mode()** API with new optional output parameter
- Wow (Benefit / Value, Need Addressed)
  - Reduce the need for recreates and other time-consuming debugging activities by providing a first failure data capture during normal execution
  - New diagnostic information provides better insight to the cause of failure rather than just a return code
  - Collected diagnostics are provided directly to exploiting applications – allowing them to fold this information into their own diagnostic mechanisms

Usage & Invocation

- System SSL has enhanced the certificate validation and handshake processes to now collect diagnostic information during execution as a first failure data capture
- This diagnostic information is provided to exploiting applications that use one of the two new certificate diagnostic features
  - SSLV3/TLS applications can obtain certificate diagnostic information about the peer certificate by implementing the new **GSK_CERT_DIAGNOSTIC_CALLBACK** routine
    - This routine is provided the peer's certificate diagnostic information that was collected during the processing of the handshake CERTIFICATE message (**gsk_secure_socket_init()** API)
  - Applications that call the **gsk_validate_certificate_mode()** API directly can obtain this diagnostic information with a new optional parameter

Usage & Invocation – Diagnostic Callback

- SSLV3/TLS applications looking to exploit the **gsk_cert_diagnostic_callback** simply need to implement their own callback routine using the provided function prototype:

```
void * cert_diagnostic_callback (
    gsk_handle                soc_handle,
    gsk_diag_summary *        diag_summary,
    char *                    diag_string,
    char *                    user_data);
```

- Then set the **GSK_CERT_DIAGNOSTIC_CALLBACK** with *gsk_attribute_set_callback()* :

```
gsk_attribute_set_callback(handle,
                           GSK_CERT_DIAGNOSTIC_CALLBACK,
                           (void *)cert_diagnostic_callback);
```

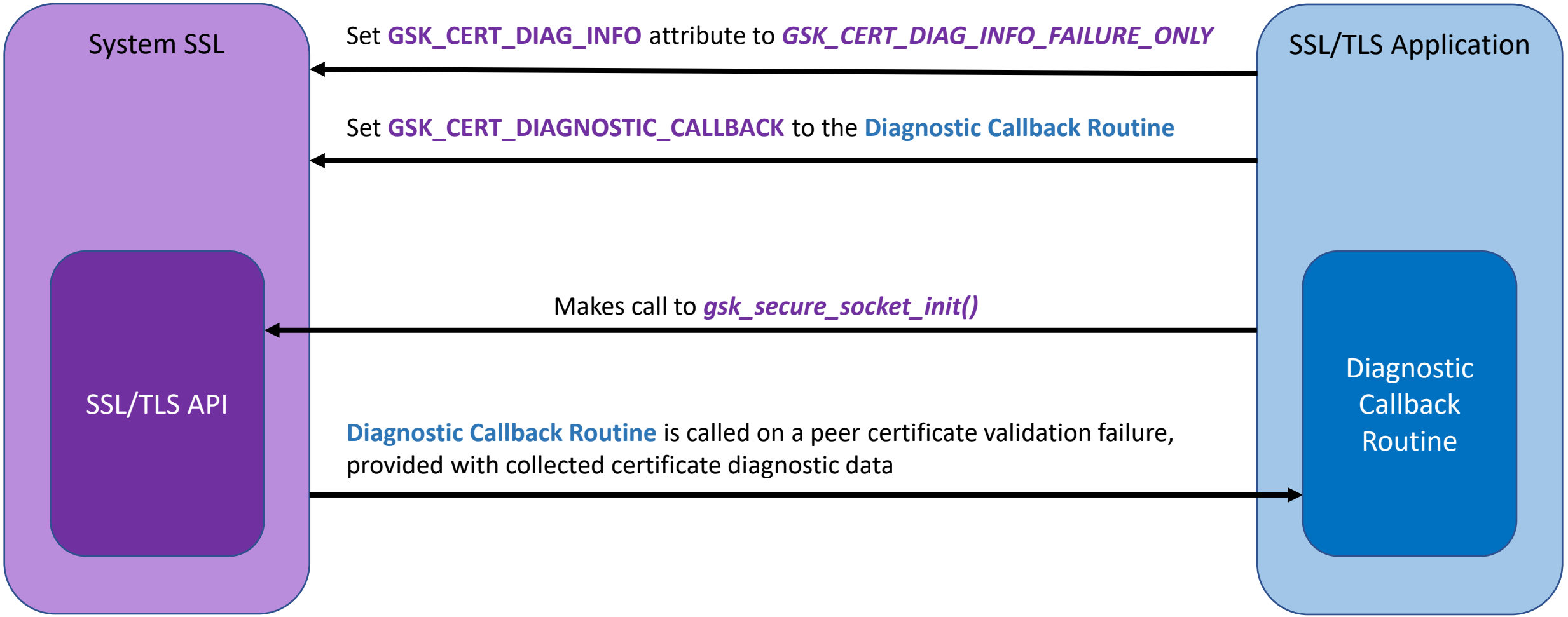
Usage & Invocation – Diagnostic Callback

- The callback routine will automatically be called whenever a certificate validation occurs depending on the setting of the **GSK_CERT_DIAG_INFO** attribute
- **GSK_CERT_DIAG_INFO** can be one of the following values:
  - ***GSK_CERT_DIAG_INFO_FAILURE_ONLY***
  - ***GSK_CERT_DIAG_INFO_SUCCESS_ONLY***
  - ***GSK_CERT_DIAG_INFO_SUCCESS_OR_FAILURE***
- By default, the callback routine is only called on validation failures
- The value for this attribute is set with either the **GSK_CERT_DIAG_INFO** environment variable or with the ***gsk_attribute_set_enum()*** routine

Usage & Invocation – Diagnostic Callback

Attribute	Description	Values	Comments
GSK_CERT_DIAG_INFO (New)	Specifies the circumstances in which the ***gsk_cert_diagnostic_callback*** routine should be called.	Environment variable allowed settings: • **FAILURE** – callback will only be called if the certificate validation fails for the peer • **SUCCESS** – callback will only be called if the certificate validation is successful for the peer • **BOTH** – callback will be called for both peer certificate validation successes and failures ***gsk_attribute_[sg]et_enum()*** allowed settings: • **GSK_CERT_DIAG_INFO_FAILURE_ONLY** • **GSK_CERT_DIAG_INFO_SUCCESS_ONLY** • **GSK_CERT_DIAG_INFO_SUCCESS_OR_FAILURE**	Default: **FAILURE**

Usage & Invocation – Diagnostic Callback



Usage & Invocation – Diagnostic Callback

- When the diagnostic callback routine is called, it is provided the peer's certificate validation diagnostic information within the ***gsk_diag_summary***

- The ***gsk_diag_summary*** provides the following information:
  - CMS & SSL Return Codes
  - A brief description (text) of the failure encountered
  - List of the certificate sources used during validation
  - Certificate details of each certificate in the certificate chain
    - SubjectDN, IssuerDN, SerialNum, etc.
    - Source that the certificate was found (Handshake, kdb file, SAF keyring, etc.)
  - Index of the failing certificate (within the certificate chain)

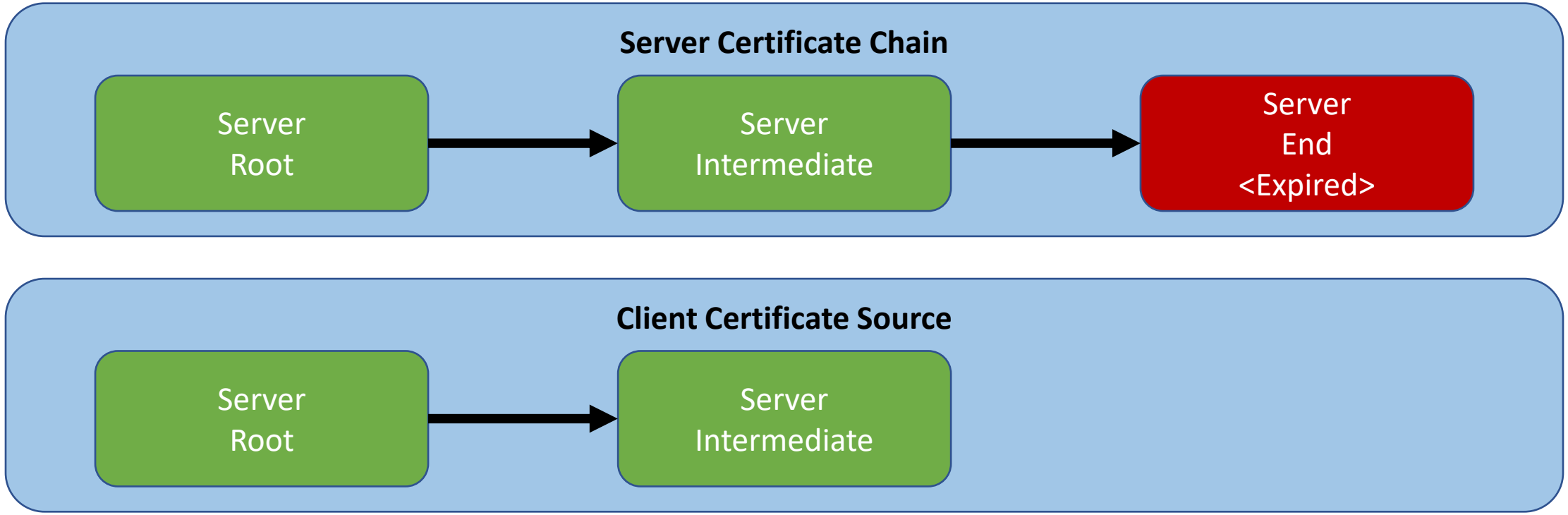
- In addition to the ***gsk_diag_summary***, the routine is also provided an untranslated diagnostic string

Usage & Invocation – Diagnostic Callback Examples

- Diagnostic Callback Example Scenarios:
  - Expired Certificate
  - Missing Root CA
  - Revoked Certificate
  - Unsupported Elliptic Curve
- N.B. The following examples use output taken from internal applications. The output and formatting of the diagnostic data shown are a result of our implementation of a sample diagnostic callback routine in order to demonstrate the data provided by the diagnostics.

Usage & Invocation – Diagnostic Callback Example 1

Example: Server's End Certificate is Expired



Usage & Invocation – Diagnostic Callback Example 1

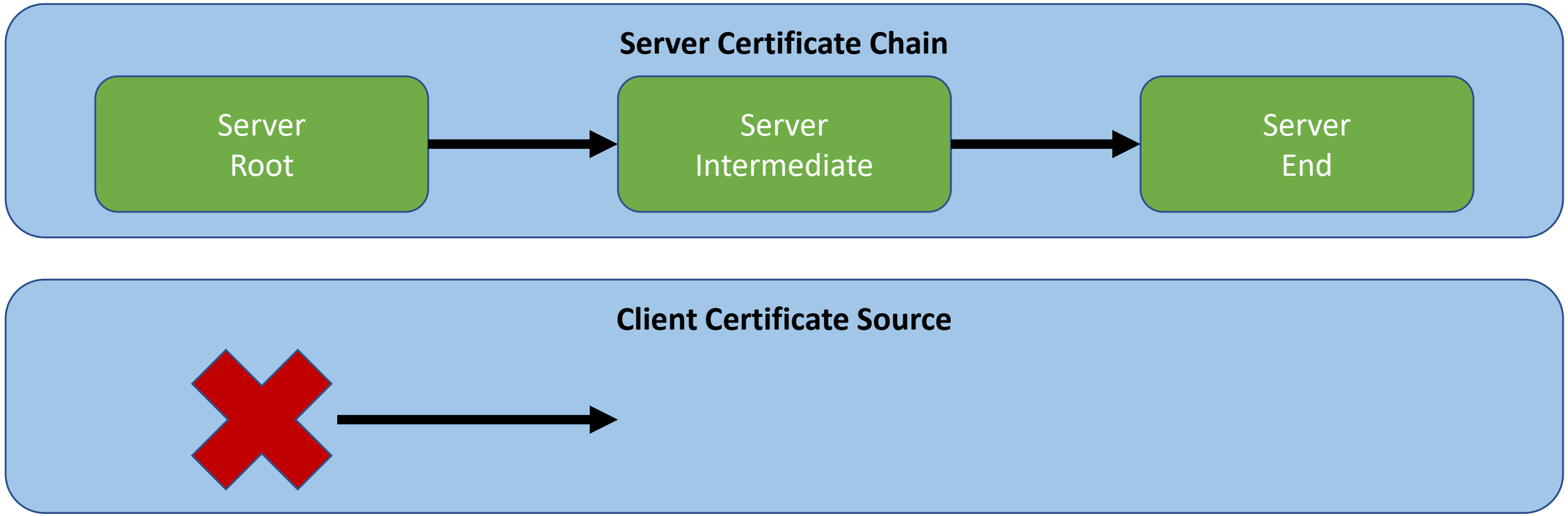
```
================================================================================
|                            Certificate Diagnostics                           |
================================================================================
|                                Summary Overview                              |
+--------------------+-----------------------------------------------------+
| SSL Return Code    |  401                                                |
| CMS Return Code    |  0x03353022                                         |
| Descriptive Text   |  Certificate is expired                             |
+--------------------+-----------------------------------------------------+
|                                Certificate Chain                             |
+--------------------+-----------------------------------------------------+
| Certificate Count  |  1                                                  |
| Failing Cert Index |  1                                                  |
+--------------------+-----------------------------------------------------+
| Certificate Index  |  1                                                  |
| SubjectDN          |  CN=End,OU=SSL,O=IBM,L=POK,ST=NY,C=US               |
| IssuerDN           |  CN=Int,OU=SSL,O=IBM,L=POK,ST=NY,C=US               |
| Serial Number      |  0d                                                 |
| Cert Source        |  Handshake                                          |
```

Usage & Invocation – Diagnostic Callback Example 1

```
+-------------------------------------------------------------------------+
|                              Diagnostic String                          |
+-------------------------------------------------------------------------+
| SSLRetCode= 401 CMSRetCode= 0x03353022 Description= Certificate is expired S |
| ubjectDN= <CN=End,OU=SSL,O=IBM,L=POK,ST=NY,C=US> IssuerDN= <CN=Int,OU=SSL,O= |
| IBM,L=POK,ST=NY,C=US> SerialNumber= 0d CertificateSource= Handshake TrustedS |
| ource= /home/certs/ex1.kdb                                              |
===========================================================================
```

Usage & Invocation – Diagnostic Callback Example 2

Example: Client is Missing Server's Root Certificate



Usage & Invocation – Diagnostic Callback Example 2

```
==========================================================================
|                          Certificate Diagnostics                       |
==========================================================================
|                             Summary Overview                           |
+--------------------+-----------------------------------------------+
| SSL Return Code    |  8                                            |
| CMS Return Code    |  0x0335302f                                   |
| Descriptive Text   |  Self-signed certificate not found in trusted key      |
|                    |  source                                       |
+--------------------+-----------------------------------------------+
|                             Certificate Chain                          |
+--------------------+-----------------------------------------------+
| Certificate Count  |  3                                            |
| Failing Cert Index |  3                                            |
```

Usage & Invocation – Diagnostic Callback Example 2

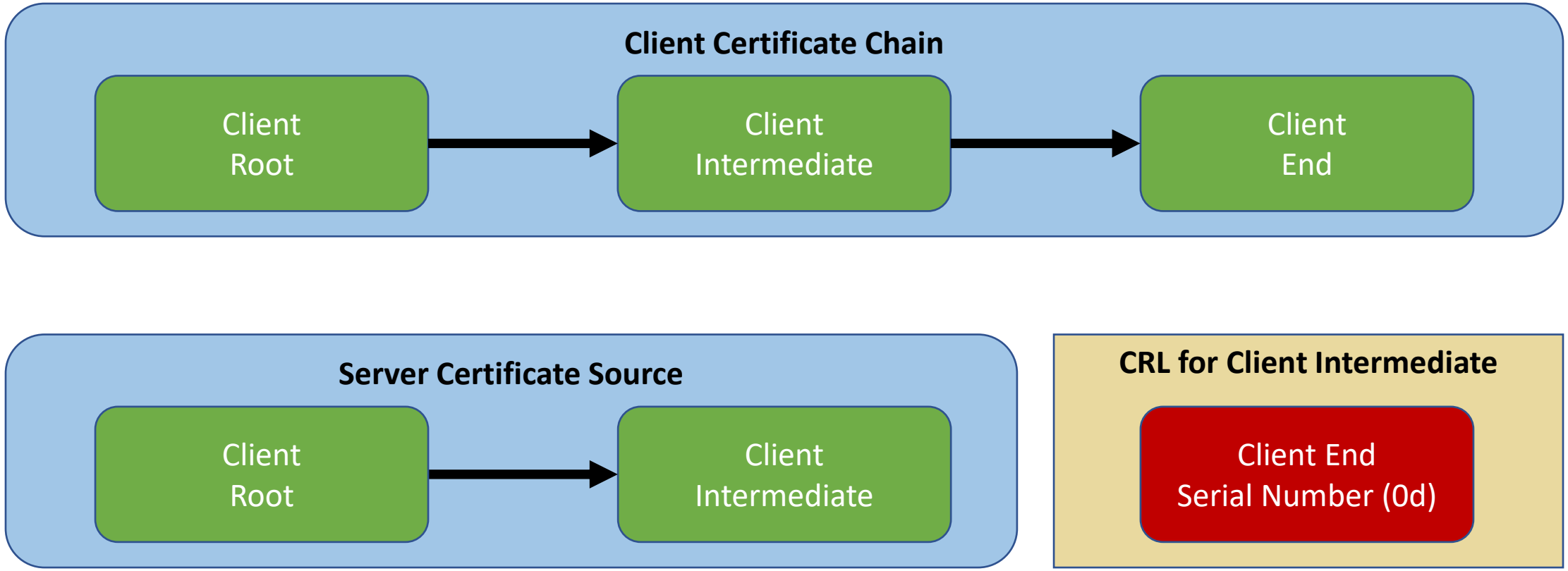
```
+--------------------+-------------------------------------------------------+
| Certificate Index  |  1                                                    |
| SubjectDN          |  CN=End,OU=SSL,O=IBM,L=POK,ST=NY,C=US                 |
| IssuerDN           |  CN=Int,OU=SSL,O=IBM,L=POK,ST=NY,C=US                 |
| Serial Number      |  0d                                                   |
| Cert Source        |  Handshake                                            |
+--------------------+-------------------------------------------------------+
| Certificate Index  |  2                                                    |
| SubjectDN          |  CN=Int,OU=SSL,O=IBM,L=POK,ST=NY,C=US                 |
| IssuerDN           |  CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US                |
| Serial Number      |  0c                                                   |
| Cert Source        |  Handshake                                            |
+--------------------+-------------------------------------------------------+
| Certificate Index  |  3                                                    |
| SubjectDN          |  CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US                |
| IssuerDN           |  CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US                |
| Serial Number      |  0b                                                   |
| Cert Source        |  Handshake                                            |
+--------------------+-------------------------------------------------------+
```

Usage & Invocation – Diagnostic Callback Example 2

```
+-------------------------------------------------------------------------+
|                              Diagnostic String                          |
+-------------------------------------------------------------------------+
| SSLRetCode= 8 CMSRetCode= 0x0335302f Description= Self-signed certificate no |
| t found in trusted key source SubjectDN= <CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C |
| =US> IssuerDN= <CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US> SerialNumber= 0b Cert |
| ificateSource= Handshake TrustedSource= /home/certs/ex2.kdb             |
=========================================================================
```

Usage & Invocation – Diagnostic Callback Example 3

Example: Client Using Revoked Certificate



Usage & Invocation – Diagnostic Callback Example 3

```
==========================================================================
|                          Certificate Diagnostics                       |
==========================================================================
|                             Summary Overview                           |
+--------------------+-----------------------------------------------+
| SSL Return Code    |  431                                          |
| CMS Return Code    |  0x03353041                                   |
| Descriptive Text   |  Using CDP HTTP CRL, certificate is revoked   |
+--------------------+-----------------------------------------------+
|                             Certificate Chain                          |
+--------------------+-----------------------------------------------+
| Certificate Count  |  3                                            |
| Failing Cert Index |  1                                            |
```

Usage & Invocation – Diagnostic Callback Example 3

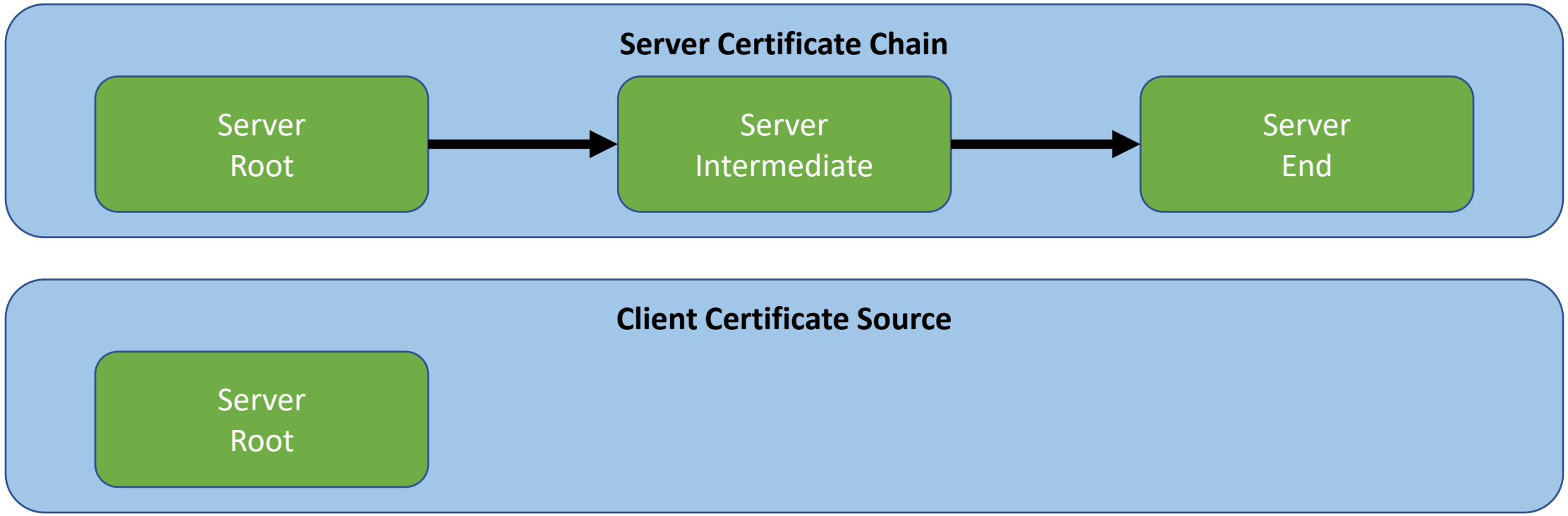
```
+--------------------+-----------------------------------------------------+
| Certificate Index  |  1                                                  |
| SubjectDN          |  CN=End,OU=SSL,O=IBM,L=POK,ST=NY,C=US               |
| IssuerDN           |  CN=Int,OU=SSL,O=IBM,L=POK,ST=NY,C=US               |
| Serial Number      |  0d                                                 |
| Cert Source        |  Handshake                                          |
+--------------------+-----------------------------------------------------+
| Certificate Index  |  2                                                  |
| SubjectDN          |  CN=Int,OU=SSL,O=IBM,L=POK,ST=NY,C=US               |
| IssuerDN           |  CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US              |
| Serial Number      |  0c                                                 |
| Cert Source        |  /home/certs/ex3.kdb                                |
+--------------------+-----------------------------------------------------+
| Certificate Index  |  3                                                  |
| SubjectDN          |  CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US              |
| IssuerDN           |  CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US              |
| Serial Number      |  0b                                                 |
| Cert Source        |  /home/certs/ex3.kdb                                |
+--------------------+-----------------------------------------------------+
```

Usage & Invocation – Diagnostic Callback Example 3

```
+-------------------------------------------------------------------------+
|                              Diagnostic String                          |
+-------------------------------------------------------------------------+
| SSLRetCode= 431 CMSRetCode= 0x03353041 Description= Using CDP HTTP CRL, cert |
| ificate is revoked SubjectDN= <CN=End,OU=SSL,O=IBM,L=POK,ST=NY,C=US> IssuerD |
| N= <CN=Int,OU=SSL,O=IBM,L=POK,ST=NY,C=US> SerialNumber= 0d CertificateSource |
| = Handshake TrustedSource= /home/certs/ex3.kdb                          |
=========================================================================
```

Usage & Invocation – Diagnostic Callback Example 4

Example: Server Using Elliptic Curve not Supported by GSK_CLIENT_ECURVE_LIST



Usage & Invocation – Diagnostic Callback Example 4

Note: The CMS return code indicates a perfectly valid certificate

```
================================================================================
|                          Certificate Diagnostics                             |
================================================================================
|                              Summary Overview                                |
+--------------------+-----------------------------------------------------+
| SSL Return Code    |  405                                                |
| CMS Return Code    |  0x00000000                                         |
| Descriptive Text   |  Elliptic curve type (secp192r1) is not supported by|
|                    |  the local GSK_CLIENT_ECURVE_LIST                   |
+--------------------+-----------------------------------------------------+
|                              Certificate Chain                               |
+--------------------+-----------------------------------------------------+
| Certificate Count  |  3                                                  |
| Failing Cert Index |  1                                                  |
```

Usage & Invocation – Diagnostic Callback Example 4

```
+--------------------+-------------------------------------------------+
| Certificate Index  |  1                                              |
| SubjectDN          |  CN=End,OU=SSL,O=IBM,L=POK,ST=NY,C=US           |
| IssuerDN           |  CN=Int,OU=SSL,O=IBM,L=POK,ST=NY,C=US           |
| Serial Number      |  0d                                             |
| Cert Source        |  Handshake                                      |
+--------------------+-------------------------------------------------+
| Certificate Index  |  2                                              |
| SubjectDN          |  CN=Int,OU=SSL,O=IBM,L=POK,ST=NY,C=US           |
| IssuerDN           |  CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US          |
| Serial Number      |  0c                                             |
| Cert Source        |  Handshake                                      |
+--------------------+-------------------------------------------------+
| Certificate Index  |  3                                              |
| SubjectDN          |  CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US          |
| IssuerDN           |  CN=Root,OU=SSL,O=IBM,L=POK,ST=NY,C=US          |
| Serial Number      |  0b                                             |
| Cert Source        |  /home/certs/ex4.kdb                            |
+--------------------+-------------------------------------------------+
```

Usage & Invocation – Diagnostic Callback Example 4

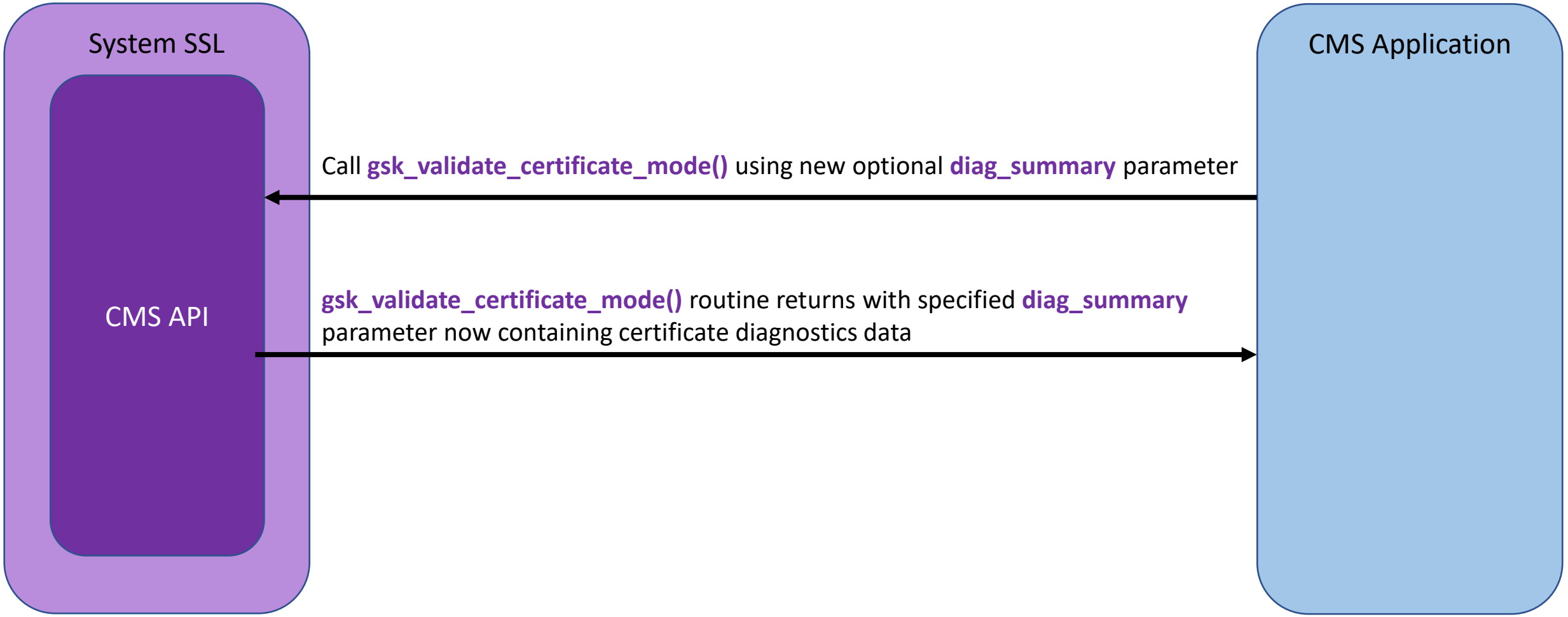
```
+-------------------------------------------------------------------------+
|                              Diagnostic String                          |
+-------------------------------------------------------------------------+
| SSLRetCode= 405 CMSRetCode= 0x00000000 Description= Elliptic curve type (sec |
| p192r1) is not supported by the local GSK_CLIENT_ECURVE_LIST SubjectDN= <CN= |
| End,OU=SSL,O=IBM,L=POK,ST=NY,C=US> IssuerDN= <CN=Int,OU=SSL,O=IBM,L=POK,ST=N |
| Y,C=US> SerialNumber= 0d CertificateSource= Handshake TrustedSource= /home/c |
| erts/ex4.kdb                                                            |
===========================================================================
```

Usage & Invocation – gsk_validate_certificate_mode() API

- Certificate diagnostics are also available to callers of the ***gsk_validate_certificate_mode()*** API through a new optional parameter

```
gsk_status gsk_validate_certificate_mode (
    gskdb_data_sources *                    data_sources,
    x509_certificate *                      subject_certificate,
    gsk_boolean                             accept_root,
    gsk_int32 *                             issuer_record_id,
    GSKCMS_CERT_VALIDATION_MODE             validation_mode,
    gsk_uint32                              arg_count
    [,GSKCMS_CERT_VALIDATE_KEYRING_ROOT     validate_root]
    [,GSKCMS_REVOCATION_SECURITY_LEVEL      security_level]
    [,gsk_int32                             max_source_rev_ext_loc_values]
    [,gsk_int32                             max_validation_rev_ext_loc_values]
    [,x509_diag_summary *                   diag_summary]
    ...)
```

Usage & Invocation – gsk_validate_certificate_mode() API



Usage & Invocation – gsk_validate_certificate_mode() API

- System SSL will collect and return certificate diagnostics if a non-NULL value is provided for the ***diag_summary*** optional parameter.

- Diagnostics are always collected and returned using this method. (Not determined by **GSK_CERT_DIAG_INFO** attribute)

- The diagnostics returned consist of:
  - CMS return code
  - A brief description (text) of the failure encountered
  - Copy of each certificate in the chain
  - Index of the failing certificate (within the certificate chain)
  - Index and type of each certificate source
  - Failing revocation source index

Interactions & Dependencies

- Software Dependencies
  - None
- Hardware Dependencies
  - None
- Exploiters
  - AT-TLS (Application Transparent –TLS)
  - IPSec

Interactions & Dependencies

- Certificate Diagnostics: AT-TLS support
- AT-TLS implements the new GSK_CERT_DIAGNOSTIC_CALLBACK callback function
- Whenever System SSL returns an error regarding a certificate received from a remote communication partner:
  - If the AT-TLS trace level includes 2 (Error) for the relevant AT-TLS rule, a certificate diagnostic message will be written to the AT-TLS log via syslogd:

```
EZD2052I TTLS Certificate Diagnostics GRPID:00000004 ENVID: 00000004 CONNID: 00000039 SSLRetCode= 8
CMSRetCode= 0x0335302f Description= Self-signed certificate not found in trusted key source SubjectDN=
<CN=TEST Server,OU=MYDEPT,O=MYCOMPANY,L=Raleigh,ST=NC,C=US> IssuerDN= <CN=TEST INTERMEDIARY
CA,OU=MYDEPT,O=MYCOMPANY,L=Raleigh,ST=NC,C=US> SerialNumber= 67 CertificateSource= Handshake TrustedSource=
CLIENTRING
```

  - If the AT-TLS trace level includes 8 (Event) for the relevant AT-TLS rule, the certificate chain will also be written to syslogd:

```
EZD2053I TTLS Certificate Diagnostics Details GRPID:00000004 ENVID: 00000004 CONNID: 00000039 Certificate=
1 of 3 FailingCert= NO SubjectDN= <CN=TEST Server,OU=MYDEPT,O=MYCOMPANY,L=Raleigh,ST=NC,C=US> IssuerDN=
<CN=TEST INTERMEDIARY CA,OU=MYDEPT,O=MYCOMPANY,L=Raleigh,ST=NC,C=US> SerialNumber= 67 CertificateSource=
Handshake
EZD2053I TTLS Certificate Diagnostics Details GRPID:00000004 ENVID: 00000004 CONNID: 00000039 Certificate=
2 of 3 FailingCert= NO  SubjectDN= <CN=TEST INTERMEDIARY CA,OU=MYDEPT,O=MYCOMPANY,L=Raleigh,ST=NC,C=US>
IssuerDN= <CN=TEST ROOT CA,OU=MYDEPT,O=MYCOMPANY,L=Raleigh,ST=NC,C=US> SerialNumber= 23 CertificateSource=
Handshake
EZD2053I TTLS Certificate Diagnostics Details GRPID:00000004 ENVID: 00000004 CONNID: 00000039 Certificate=
3 of 3 FailingCert= YES  SubjectDN= <CN=TEST ROOT CA,OU=MYDEPT,O=MYCOMPANY,L=Raleigh,ST=NC,C=US> IssuerDN=
<CN=TEST ROOT CA,OU=MYDEPT,O=MYCOMPANY,L=Raleigh,ST=NC,C=US> SerialNumber= 11 CertificateSource= Handshake
EZD2054I TTLS Certificate Diagnostics Data Sources GRPID:00000004 ENVID: 00000004 CONNID: 00000039 Count= 2
CLIENTRING , Handshake
```

Upgrade & Coexistence Considerations

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

- Migration/Toleration/Coexistence
  - None

Installation & Configuration

- None

Extended Master Secret

Overview

- Who (Audience)
  - Applications performing TLS 1.0, TLS 1.1 and TLS 1.2 handshakes
- What (Solution)
  - Implement the extended master secret (EMS) support as specified by RFC 7627
- Wow (Benefit / Value, Need Addressed)
  - Uses an improved and more secure calculation for the master secret during TLS 1.0, TLS 1.1, and TLS 1.2 handshakes

Usage & Invocation

- System SSL has added EMS support in z/OS 2.5
- EMS support will be available in the future for z/OS 2.3 and 2.4 with new function APAR OA60105
- By default, System SSL will enable EMS support on the client and server with addition of two new environment variables/attributes
  - This results in the negotiation of EMS with the client and server by default
- EMS is only negotiated during TLS 1.0, TLS 1.1, and TLS 1.2 handshakes
  - It does not apply to TLS 1.3 as an EMS-styled calculation is already done within this protocol

Usage & Invocation

Attribute	Description	Values	Comments
GSK_CLIENT_EXTENDED_MASTER_SECRET (New)	Specifies if the TLS client sends the extended master secret extension to the server. This option is only applicable for TLS V1.0, TLS V1.1, or TLS V1.2 handshakes.	Environment variable allowed settings: • 0, OFF, or DISABLED – Does not send the extended master secret extension to the server • 1, ON, or ENABLED – Sends the extended master secret extension to the server but does not require the server to support the extension • REQUIRED – Sends the extended master secret extension to the server and requires the server to support the extension. If the server does not send the extension, the handshake fails. gsk_attribute_[sg]et_enum() allowed settings (connection or environment): • GSK_CLIENT_EXTENDED_MASTER_SECRET_ON • GSK_CLIENT_EXTENDED_MASTER_SECRET_OFF • GSK_CLIENT_EXTENDED_MASTER_SECRET_REQUIRED	Default: ON

Usage & Invocation

Attribute	Description	Values	Comments
GSK_SERVER_EXTENDED_MASTER_SECRET (New)	Specifies if the TLS server supports negotiating the extended master secret extension from clients. This option is only applicable for TLS V1.0, TLS V1.1, or TLS V1.2 handshakes.	Environment variable allowed settings: • 0, OFF, or DISABLED – Does not support negotiating the extended master secret extension from clients • 1, ON, or ENABLED – Supports negotiating the extended master secret extension from clients but does not require the extension • REQUIRED – Requires the EMS extension from the client. If a client does not send the extension, the handshake fails. gsk_attribute_[sg]et_enum() allowed settings (connection or environment): • GSK_SERVER_EXTENDED_MASTER_SECRET_ON • GSK_SERVER_EXTENDED_MASTER_SECRET_OFF • GSK_SERVER_EXTENDED_MASTER_SECRET_REQUIRED	Default: ON

Usage & Invocation

- gsk_attribute_get_enum() – Updated to support a new attribute which can be queried to see if the EMS extension has been negotiated on a connection
  - New attribute type: GSK_EXTENDED_MASTER_SECRET_USED
  - GSK_EXTENDED_MASTER_SECRET_USED_ON – Indicates that EMS has been negotiated on the connection
  - GSK_EXTENDED_MASTER_SECRET_USED_OFF – Indicates EMS has not been negotiated on the connection

Interactions & Dependencies

- Software Dependencies
  - None
- Hardware Dependencies
  - None
- Exploiters
  - AT-TLS (Application Transparent – Transport Layer Security)

Upgrade & Coexistence Considerations

- To exploit this solution, all systems in the Sysplex must be at the new z/OS level: Yes (for now)
  - APAR OA60105 provides full EMS negotiation for LPARs running 2.3 and 2.4
  - When the PTFs for OA60105 are applied, then all systems will have the capability to negotiate EMS (full and cached TLS handshakes)
  - 2.3 PTFs (UJ05345, UJ05349, and UJ05361) and 2.4 PTFs (UJ05348, UJ05368, and UJ05370)
- Toleration/coexistence APAR OA60691
  - Allows for cached TLS handshakes to occur on back-level LPARs that negotiate EMS on 2.5
  - Only comes into play when the System SSL server application is enabled for sysplex caching (GSK_SYSPLEX_SIDCACHE=ON)
  - Before IPLing z/OS 2.5, all LPARs in the sysplex must apply the PTFs for coexistence APAR OA60691 to all back-level releases (2.3 and 2.4)
    - If the PTFs are not applied, additional full TLS handshakes may occur which may impact performance
    - Coexistence APAR OA60691 will be marked as IBM.Coexistence.z/OS.V2R5
  - 2.3 PTFs (UJ05161 and UJ05173) and 2.4 PTFs (UJ05162 and UJ05195)

Upgrade & Coexistence Considerations

- Updates in z/OS 2.5
  - Server will now optionally negotiate EMS if the client has sent the extension
  - Client will now send and optionally negotiate EMS if the server supports it
  - Can set GSK_CLIENT_EXTENDED_MASTER_SECRET and GSK_SERVER_EXTENDED_MASTER_SECRET to OFF to turn off negotiating EMS

Installation & Configuration

- None

Summary

- You should now be able to understand the following enhancements from System SSL:
  - Certificate failure diagnostics
  - Extended master secret support
  - Understand how these enhancements affect installation, migration and coexistence

Appendix

- z/OS Cryptographic Services System SSL Programming