SMF Digital Signatures in z/OS 2.2

Anthony Sofia (atsofia@us.ibm.com)

Senior Software Engineer at IBM

NY RUG Meeting – Nov 20th 2015

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

CICS*
DB2*
DS8000*
IBM*
IBM eServer
IBM (logo)*
IBM Sterling Connect:Direct*
MQSeries*
RMF
System z*
zEnterprise*
z/OS*
zSeries*

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

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 therefrom.

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.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

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.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and

Linux is a registered trademark of Linus Torvalds 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.

OpenStack is a trademark of OpenStack LLC. The OpenStack trademark policy is available on the OpenStack website.

TEALEAF is a registered trademark of Tealeaf, an IBM Company.

Windows Server and the Windows logo are trademarks of the Microsoft group of countries.

Worklight is a trademark or registered trademark of Worklight, an IBM Company.

UNIX is a registered trademark of The Open Group in the United States and other countries.

* Other product and service names might be trademarks of IBM or other companies.

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.

This information provides only general descriptions of the types and portions of workloads that are eligible for execution on Specialty Engines (e.g, zIIPs, zAAPs, and IFLs) ("SEs"). IBM authorizes customers to use IBM SE only to execute the processing of Eligible Workloads of specific Programs expressly authorized by IBM as specified in the “Authorized Use Table for IBM Machines” provided at www.ibm.com/systems/support/machine_warranties/machine_code/aut.html (“AUT”). No other workload processing is authorized for execution on an SE. IBM offers SE at a lower price than General Processors/Central Processors because customers are authorized to use SEs only to process certain types and/or amounts of workloads as specified by IBM in the AUT.

Agenda

- What is a digital signature?
- How digital signatures enhance SMF data
- Configuration and Usage

What is a Asymmetric Cryptography

- Also known as Public-Key Cryptography
- Used for message encryption (i.e. to transmit a key for symmetric encryption) or for message signatures
- Utilizes very large random numbers – Strength lies in the inability to factor these very large numbers
- The term "asymmetric" comes from the use of different keys, a public key and private key, to perform these opposite functions

What is a Digital Signature?

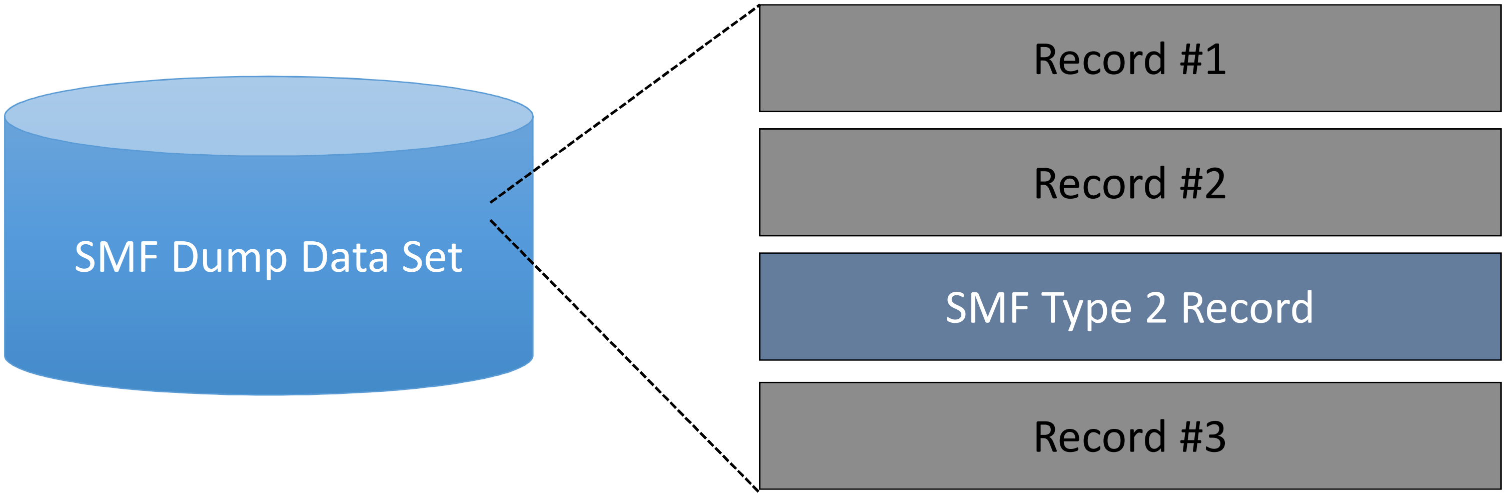
- A way to ensure the source and validity of data
- The signer will first hash the data and then encrypt the hash with their ***private key*** – The encrypted hash is the ***signature***
- The consumer of the data can hash the same data and decrypt the signature, using the ***public key,*** to obtain the signer's hash
- The hashes will then be compared – When these values match then the data contents and source are verified

Storing SMF Digital Signatures

- Digital Signatures are stored in SMF2 records
  - *Subtype 1 provides a grouped signatures*
  - *Subtype 2 provides interval based signatures*
  - *Data must be validated on interval boundaries*
- New data included in these records includes counts of records included, start and end times of the data included and the hashing and signature methods
- SMF2 records today are generated by IFASMFDL and IFASMFDP and is ignored by these utilities by default

Storing SMF Digital Signatures (cont)

- Looking at a data set dumped from a logstream the SMF Type 2 records will be integrated into the data



When SMF Signs Records

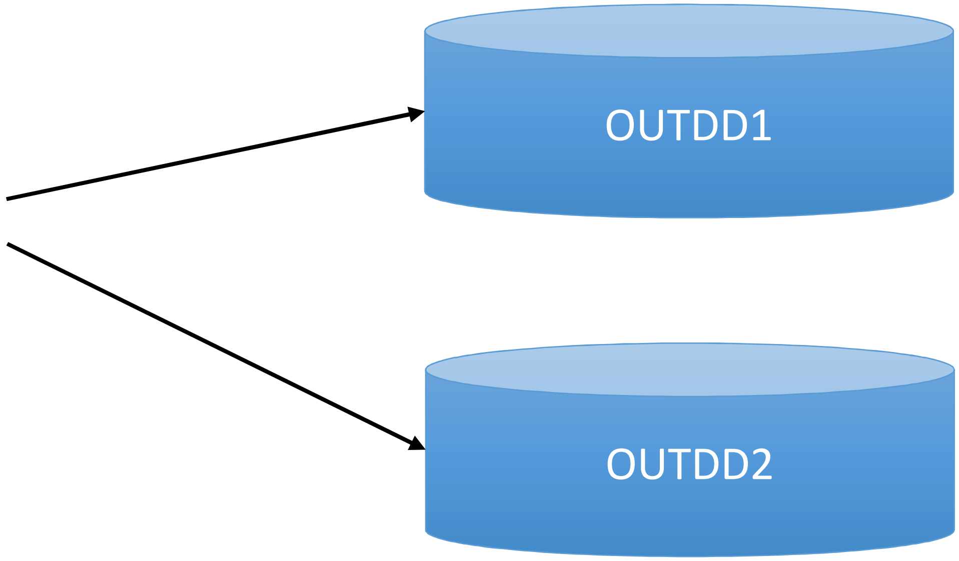
- The SMF data is signed on the way to System Logger
  - This function is only available when using SMF Logstream Recording!
  - As each block of records is written to the logstream
    - Each record is individually hashed
    - Running hash maintained per unique SMF type/subtype
- Periodically, the hash will be encrypted and the digital signature data will be recorded to the logstream as a ***group signature record***
- On the global interval a signature is created for all data hashed since the previous interval and recorded to the logstream as an ***interval signature record***
- These operations are performed with the ***private key***

When SMF Moves Signature Records

- IFASMFDL and IFASMFDP understand signature records
- Both utilities can optionally carry them to an OUTDD data set with the records of an associated SMF type/subtype
  - OUTDD data sets can be independently verified

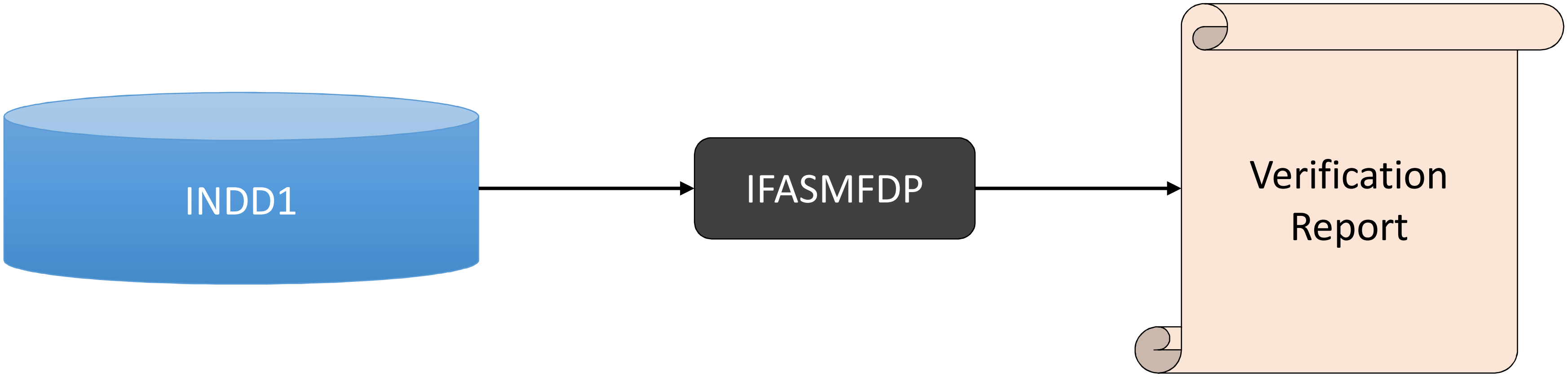
IFASMFDP or IFASMFDL SYSIN

```
OUTDD(OUTDD1,TYPE(23))
OUTDD(OUTDD2,TYPE(30))
```



When SMF Verifies Records

- IFASMFDP can verify a set of SMF records has not been tampered with when signature records are available.
    - The Signature Records must have been carried through successive IFASMFDL and IFASMFDP passes over the data



Setup Steps

- The first step is create a public/private key pair via ICSF
  - SMF does not care about the type of key (clear or secure) as long as the available hardware can support it
- Scope of the key usage can be per enterprise, sysplex, system or logstream
- SMF needs the token name to perform the PKCS#11 functions via ICSF as well as the type of encryption – For example RSA or Elliptical Curve
- The SMF address space and any invokers of IFASMFDP will need access to ICSF, PKCS#11 and the appropriate key
  - See SAF resources CRYPTOZ, CSFSERV and CSFKEYS

Setup Steps – cont.

- Update the SMF configuration to sign record
- New option RECSIGN can be specified globally or per LSNAME
  - The logstream must be defined with a MAXBUFSIZE of 65532
- Default is NORECSIGN
- Sub-options include HASH, TOKENNAME, SIGNATURE

```
RECSIGN(HASH(SHA512),SIGNATURE(RSA),
TOKENNAME(TAMPER#RESISTANT#SMF#TOKEN#NAME1))
```

- These options are dynamic however changing these options requires some operational coordination
  - Data can only be verified with a single set of parameters, new and old data must be segregated

Setup Steps – IFASMFDL

- IFASMFDL can carry signature data with the SMF records
- By default IFASMFDL will drop signature records
  - The NOSIGSTRIP option can be used to have signature records written to OUTDD data sets
  - IFASMFDL will carry signature records transparently
- If there are multiple OUTDD statements for different types and subtypes IFASMFDL will carry the correct signature records to each OUTDD
- When signature records are carried the IFASMFDL output reports a TYPE2 record as written for each signature record

Setup Steps – IFASMFDP

- IFASMFDP can carry signature records and perform validation
- New IFASMFDP parameters NOSIGSTRIP and SIGVALIDATE
  - NOSIGSTRIP behaves the same as with IFASMFDL
- SIGVALIDATE indicates that signature validation is to be performed, Suboptions include TOKENNAME and HASH

```
SIGVALIDATE(HASH(SHA512),
TOKENNAME(TAMPER#RESISTANT#SMF#TOKEN#NAME1))
```

- Default: NOSIGVALIDATE (don't perform validation)

Setup Steps – IFASMFDP

- The relationship between PARMLIB member SMFPRMxx and the IFASMFDP options
- The TOKENNAME and HASH values must match between SMFPRMxx and IFASMFDP
- The TOKENNAME is associated with the public/private pair of keys
- IFASMFDP only needs to access the public key

SYS1.PARMLIB(SMFPRMxx)

```
LSNAME(IFASMF.xxx,TYPE(xx:yy),
RECSIGN(TOKENNAME(< 32 Char Token Name>),
SIGNATURE(yyyy),
HASH(xxx))
```

IFASMFDP SYSIN

```
SIGVALIDATE(TOKENNAME(<32 Char Token Name>),HASH(xxx))
```

IFASMFDP SIGVALIDATE Considerations

- The behavior for DATE, START and END are slightly different. Align each with an interval to ensure complete intervals of records can be validated.
- Records must retain the same order and contents as they where originally written for signature verification to succeed
- IFASMFDP ends processing after the first failure is detected

Configuration Changes

- Encryption options can be changed dynamically
  - This is not advised as it creates operational problems
  - IFASMFDP needs to be told the encryption parameters and can not validate a data set with a mix of parameters for a single SMF type/subtype from a given SID
- If options must be changed create a new logstream with the new options
  - Temporarily run with both logstreams then turn off the old logstream
  - Now there is a clean break between data signed with the old and new parameters

Configuration Changes (cont)

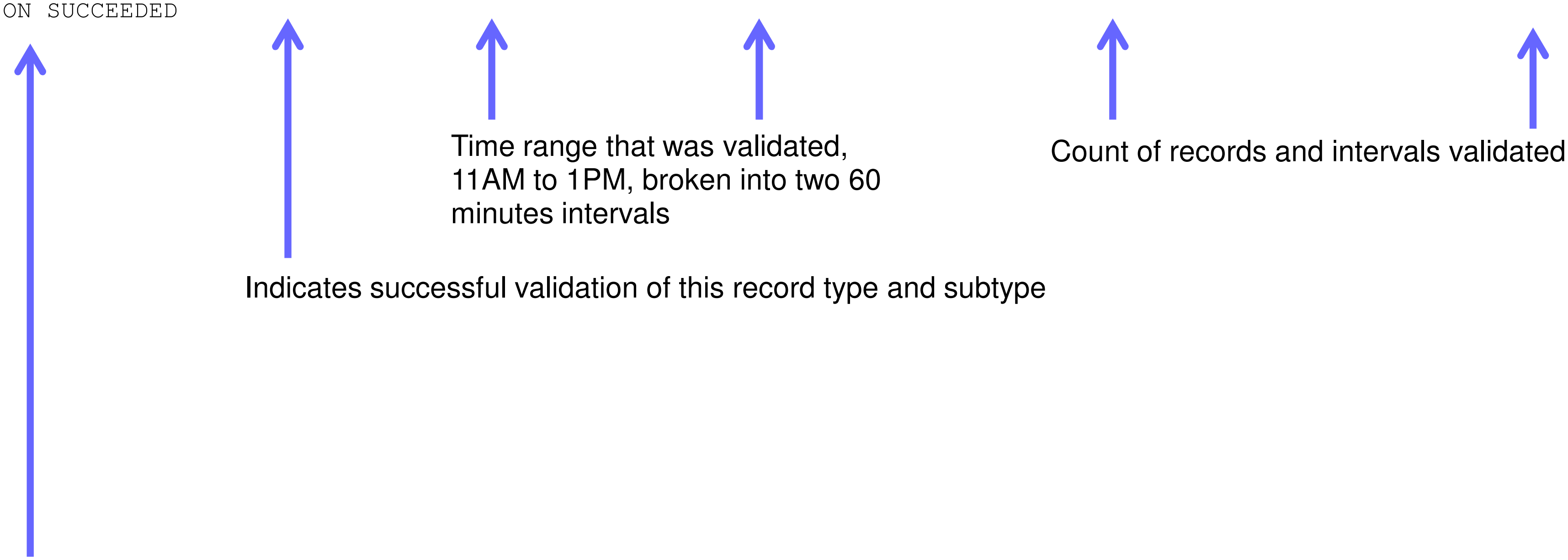
- New SMFPRMxx setting RECSIGN with options HASH, TOKENNAME, SIGNATURE apply at specific times
  - Records written before the first global interval of IPL are signed immediately
  - Records written before the first global interval of a logstream which has not been previously been signing are signed immediately
  - Records will not be signed until the global interval after a SET SMF or SETSMF command is processed for logstreams which had previously been signing

IFASMFDP Record Validation Report

- Report line generated for each SMF type and subtype processed for each SID seen
- Includes time span and counts for records that were verified
- Counts include records processed, groups processed and intervals processed
- A group is a subset of records that were signed together
- An interval is the signature generated on the SMF configured interval time
- Provides information about failures
- A signature failure is the highest level failure
- Additional checking is performed to see if the error could be due to a missing or added record or an entire missing interval of records
- Manual examination will be required to determine the root-cause of the error

Dissecting an IFASMFDP SIGVALIDATE Report

```
                           RECORD VALIDATION REPORT FOR SY1
RECORD   RECORD   VALIDATION      VALIDATION START       VALIDATION END       RECORDS        GROUPS     INTERVALS
  TYPE  SUBTYPE      FAILURE             DATE-TIME            DATE-TIME     VALIDATED     VALIDATED     VALIDATED
   128       *             N   10/23/2014-11:00:00  10/23/2014-13:00:00            60            10             2
   145       1             N   10/23/2014-11:00:00  10/23/2014-13:00:00             3             3             2
   160       *             N   10/23/2014-11:00:00  10/23/2014-13:00:00            10             2             2
VALIDATION SUCCEEDED
```



Time range that was validated, 11AM to 1PM, broken into two 60 minutes intervals

Count of records and intervals validated

Indicates successful validation of this record type and subtype

When all data validates the report ends with this message. On a failure this would provide additional information

Validation Reports – When it fails

- The report will end with VALIDATION FAILED status
- Only a single error is reported per IFASMFDP run
- IFA742I reports details about the failure

Validation Reports – IFA742I Reasons

- CRYPTOGRAPHY FAILURE - ICSF RC/RSN=<rc>/<rsn>
  - ICSF is inactive or other ICSF high level error
  - Check *Cryptographic Services ICSF Application Programmer's Guide*
- INCONSISTENT RECORDS - RECORDS DO NOT MATCH EXPECTED COUNTS
  - Potential inserted or deleted record
- INCONSISTENT RECORDS - RECORDS DO NOT MATCH EXPECTED TIMES
  - Interval record does not have the correct time relative to previous Interval record
  - A record does not have a consistent time relative to other records in the group
- INCONSISTENT RECORDS - FIRST FLAG DOES NOT MATCH
  - Group records set a first flag for the first group in each interval
  - Interval record set a first flag for the first interval written
  - Altered, inserted or deleted signature data

Validation Reports – IFA742I Reasons (cont)

- RECORD AND SUPPLIED CRYPTO OPTIONS DO NOT MATCH
  - When signature record contains different SIGVALIDATE options than IFASMFDP parameter
- MISSING RECORDS - STARTING INTERVAL
  - Started validation without initializing interval signature record and failed
  - Change your START time or possible deletion of records prior to validating the first interval record
- MISSING RECORDS - ENDING INTERVAL
  - When last interval record time does not match IFASMFDP ENDTIME parameter
  - Change END time or possible deletion of trailing records
- INCOMPLETE VALIDATION - ENDED WITH PARTIAL INTERVAL
  - Outstanding records were not validated, missing an interval record

IFASMFDL and IFASMFDP Exits

- The IFASMFDL and IFASMFDP provide an exit interface to intercept records that are processed
    - This is the USER2 exit that can be specified on the SYSIN statement
- This exit will get control for signature records that will be written to the output data set
- The SIGSTRIP option will cause these records to not be written but also will not provide them to the USER2 exit

Toleration Support

- Without enabling the new options nothing changes
- Signatures can be turned on however validation processing is not required – It is performed as needed
- At any point signatures can be stripped by IFASMFDL and IFASMFDP to provide an output data set with NO signature records
- Coexistence APAR OA47012 will provide toleration support to accept and ignore the new SMFPRMxx keywords on z/OS V1R13 and V2R1 systems

Appendix

- z/OS MVS System Management Facilities (SMF) – SA38-0667
- Z/OS MVS Initialization and Tuning Reference – SA32-0991
- z/OS Cryptographic Services ICSF Administrator's Guide - SA22-7521

Thank You!