IBM Education Assistance for z/OS V2R2

Item: Tamper Resistant SMF
Element/Component: BCP SMF

Material current as of May 2015

Agenda

- Trademarks
- Presentation Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Migration & Coexistence Considerations
- Installation
- Presentation Summary
- Appendix

Trademarks

- See URL http://www.ibm.com/legal/copytrade.shtml for a list of trademarks.

Presentation Objectives

- Things you will learn from this session
  - The purpose of Tamper Resistant SMF
    The functional benefit and content
  - How to invoke the new functionality
  - Migration / coexistence issues or concerns
  - List of publications and references

Overview

- Problem Statement
  - SMF records contain critical information about an enterprise and are archived for long durations.
  - The SMF records are generally shared among various departments for a number of activities.
  - There is no built in protection of the SMF data
- Solution
  - Use digital signatures to detect a change, addition/removal of an SMF record from a group of records.
- Value
  - Increases the value of SMF data by making it verifiable
  - Applications recording to SMF can transparently leverage this support
  - Industry standard encryption

Overview (cont.)

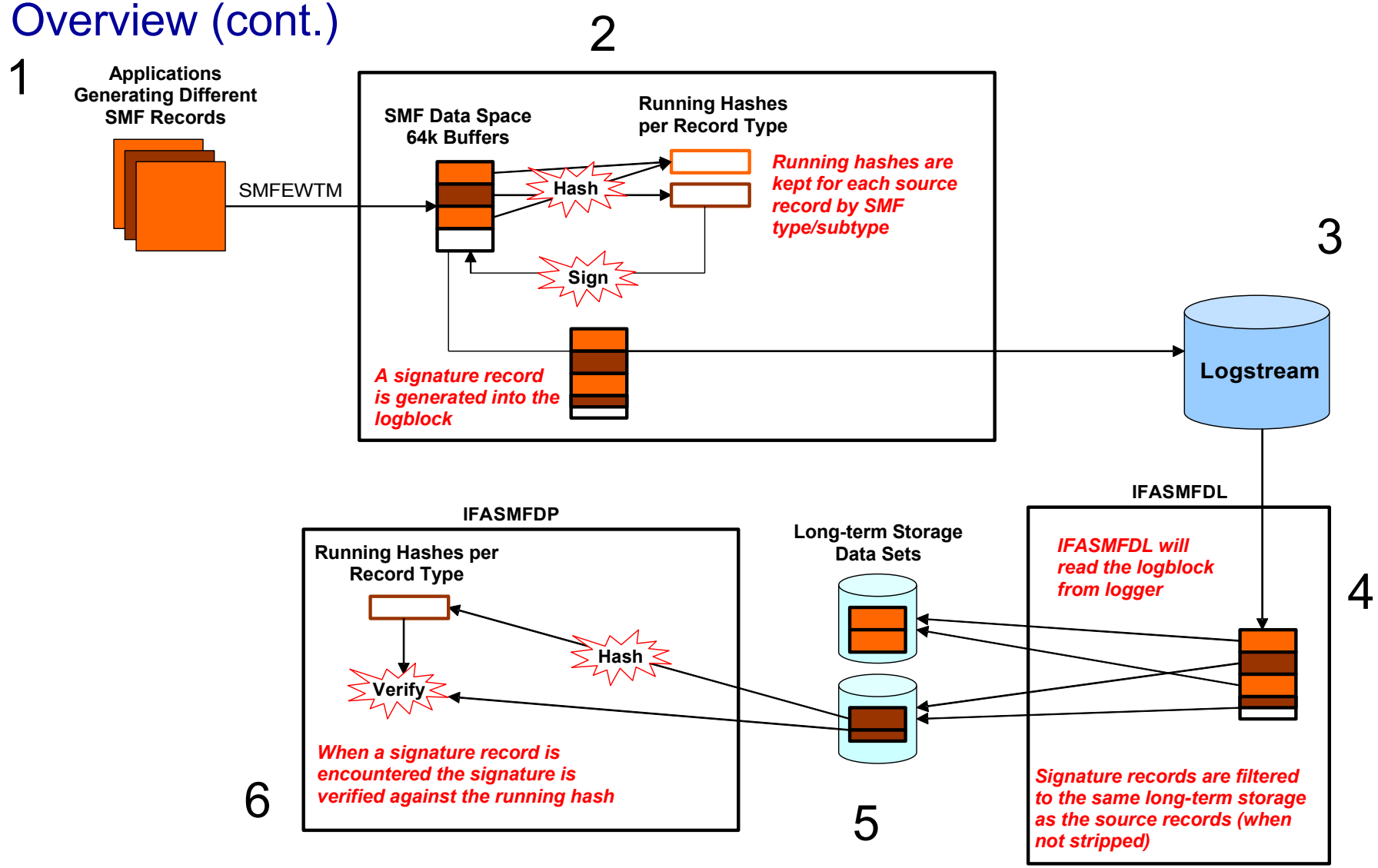
What is a digital signature?

- 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 to obtain the signer's hash
- The hashes will then be compared – When these values match then the data is considered verified
- Result – detection and deterrence of data corruption

Overview (cont.)

- The SMF data is signed on the way to System Logger
  - As each record is written to the logstream it is hashed
    - Running hash maintained per unique SMF type/subtype
  - Periodically, the hash will be digitally signed and that signature data will be recorded to the logstream as a signature record
  - On the global interval a signature is created for all data hashed during the interval and recorded to the logstream
  - These operations are performed with the private key
- IFASMFDL understands signature records and will optionally move them with the records of an associated SMF type/subtype
- IFASMFDP can verify a set of SMF records has not been tampered with when signature records are available.
  - This operation performed with the public key

Overview (cont.)



Usage & Invocation

- 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*

Usage & Invocation (cont.)

- Update the SMF configuration to sign record
  - New option *RECSIGN* can be specified globally or per *LSNAME*
  - 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

Usage & Invocation (cont.)

- 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 report a TYPE2 record as output for each signature record

Usage & Invocation (cont.)

- IFASMFDP can carry signature records and perform validation
  - New IFASMFDP parameters *SIGSTRIP* and *SIGVALIDATE*
  - *SIGSTRIP* 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)

Usage & Invocation (cont.)

- 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))
```

Usage & Invocation (cont.)

- 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

Usage & Invocation (cont.)

- The new 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

Usage & Invocation (cont.)

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
```

Indicates successful validation of this record type and subtype

Time range that was validated

Count of records and intervals validated

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

Interactions & Dependencies

- Software Dependencies
  - Integrated Cryptographic Service Facility (ICSF)
- Hardware Dependencies
  - None
- Exploiters
  - None

Migration & Coexistence Considerations

- Coexistence APAR OA47012 will provide toleration support to accept and ignore the new SMFPRMxx keywords on z/OS V1R13 and V2R1 systems

Installation

- See the previous slides for information on
  - Creating keys
  - Updating SMFPRMxx to sign SMF records
  - Updating IFASMFDL to carry signature records to data sets
  - Preparing IFASMFDP for validation

Presentation Summary

- SMF records can now be digitally signed and verified

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