Internet Printing Protocol Working Group
INTERNET DRAFT
Expires 8 February 2001

Tom Hastings
Xerox
Ira McDonald
High North
8 August 2000

Internet Printing Protocol (IPP): Notifications over SNMP via Job Monitoring MIB Traps

<draft-ietf-ipp-not-over-snmp-04.txt>

Copyright (C) The Internet Society (2000). All Rights Reserved.

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Abstract

This document is a submission to the Internet Printing Protocol Working Group of the Internet Engineering Task Force (IETF). This document is intended for eventual publication as an IETF Informational RFC. Comments should be submitted to the ipp@pwg.org mailing list.

This document specifies an OPTIONAL mapping of IPP notifications over SNMP via new service and job trap extensions to the original Job Monitoring MIB [RFC-2707]. This mapping may be used to deliver printer notifications for any printer (not just IPP-capable ones) and also to deliver job notifications for any job (not just ones submitted via IPP).

This document specifies (4) new object groups and (4) new SNMP traps for addition to the Job Monitoring MIB [RFC 2707]. A working copy of these extensions inserted in the original Job Monitoring MIB ASN.1 is

at:

ftp://ftp.pwg.org/pub/pwg/ipp/new_NOT/jmp-trap-000808.mib

Table of Contents

1. Introduction 4
2. Terminology 4
  2.1. Terminology for Conformance 4
  2.2. SNMP Network Management Framework 4
3. Model and Operation 5
  3.1. SNMP Mapping for IPP Notifications 5
    3.1.1. SNMP Mapping for IPP Printer Events 5
    3.1.2. SNMP Mapping for IPP Job Events 6
  3.2. Subscription for IPP Notifications 7
    3.2.1. Subscription via IPP 7
    3.2.2. Subscription via SNMP 7
  3.3. Relationship to other MIBs 8
    3.3.1. MIB-II (RFC 1213) 8
    3.3.2. Host Resources MIB (RFC 2790) 8
    3.3.3. Printer MIB (RFC 1759) 8
4. General Information 9
  4.1. Table 1 - Information about the Delivery Method 9
5. Subscription Template Attributes 12
  5.1. Table 2 - Subscription and Printer Attributes 12
  5.2. Additional Subscription Template Attributes 12
    5.2.1. notify-snmp-version (type2 keyword) 12
      5.2.1.1. notify-snmp-version-default 13
      5.2.1.2. notify-snmp-version-supported 13
    5.2.2. notify-snmp-auth-data (octetString(MAX)) 13
      5.2.2.1. notify-snmp-auth-data-default 14
      5.2.2.2. notify-snmp-auth-data-supported 14
    5.2.3. notify-snmp-operation (type2 keyword) 14
      5.2.3.1. notify-snmp-operation-default 15
      5.2.3.2. notify-snmp-operation-supported 15
    5.2.4. notify-snmp-mtu-size (integer(0:MAX)) 15
      5.2.4.1. notify-snmp-mtu-size-default 15
      5.2.4.2. notify-snmp-mtu-size-supported 16
  5.3. Additional Info about Subscription Template Attributes . 16
    5.3.1. notify-recipient-uri (uri) 16
6. Event Notification Content 17
  6.1. SNMP Message Envelope 17
    6.1.1. SNMPv1 Message Envelope 17
      6.1.1.1. version (INTEGER) 17
      6.1.1.2. community (OCTET STRING) 17
    6.1.2. SNMPv2 Message Envelope 17
      6.1.2.1. version (INTEGER) 17
      6.1.2.2. community (OCTET STRING) 17
    6.1.3. SNMPv3 Message Envelope 17
      6.1.3.1. msgVersion (INTEGER) 18
      6.1.3.2. msgGlobalData (HeaderData) 18
      6.1.3.3. msgSecurityParameters (OCTET STRING) 18
      6.1.3.4. msgData (ScopedPduData) 18

6.2. SNMP Message Header 18
6.2.1. SNMPv1 Trap Message Header 18

6.2.1.1. enterprise (OBJECT IDENTIFIER) 19
6.2.1.2. agent-addr (NetworkAddress) 19
6.2.1.3. generic-trap (INTEGER) 19
6.2.1.4. specific-trap (INTEGER) 19
6.2.1.5. time-stamp (TimeTicks) 19
6.2.2. SNMPv2 and SNMPv3 Trap/Inform Message Header 19
6.2.2.1. request-id (Integer32) 19
6.2.2.2. error-status (INTEGER) 19
6.2.2.3. error-index (INTEGER) 20
6.3. SNMP Message Body 20
6.3.1. SNMPv1 Trap Message Body 20
6.3.2. SNMPv2 and SNMPv3 Trap/Inform Message Body 20
6.3.2.1. sysUpTime (TimeTicks) 20
6.3.2.2. sysTrapOID (OBJECT IDENTIFIER) 20
6.4. SNMP Trap/Inform Examples 20
7. SNMP ASN.1 for IPP Notifications 21
7.1. Notification Definitions 21
7.1.1. Service Event Notify Group - Trap 21
7.1.2. Job Event Notify Group - Trap 22
7.1.3. Job Completed Notify Group - Trap 23
7.1.4. Job Progress Notify Group - Trap 24
7.2. Object Definitions 25
7.2.1. Service Group - Objects 25
7.2.2. Service Event Group - Objects 28
7.2.3. Job Event Group - Objects 32
7.2.4. Job Progress Group - Objects 35
8. Conformance 37
9. IANA Considerations 37
10. Internationalization Considerations 37
11. Security Considerations 37
12. References 38
13. Change Log 40
14. Intellectual Property Notice 43
15. Authors' Addresses 43
16. Full Copyright Statement 44

1. Introduction

This document is a submission to the Internet Printing Protocol Working Group of the Internet Engineering Task Force (IETF). Comments should be submitted to the ipp@pwg.org mailing list.

This document specifies an OPTIONAL mapping of IPP notifications over SNMP via new service and job trap extensions to the original Job Monitoring MIB [RFC-2707]. This mapping may be used to deliver printer notifications for any printer (not just IPP-capable ones) and also to deliver job notifications for any job (not just ones submitted via IPP).

This document specifies (4) new object groups and (4) new SNMP traps for addition to the Job Monitoring MIB [RFC 2707]. A working copy of these extensions inserted in the original Job Monitoring MIB ASN.1 is at:

ftp://ftp.pwg.org/pub/pwg/ipp/new_NOT/jmp-trap-000808.mib

The IPP protocol [IPP-PRO] supports passive monitoring of IPP Printer and Job objects, via client polling of IPP object attributes using the 'Get-Printer-Attributes' and 'Get-Job-Attributes' operations.

This document specifies support for dynamic monitoring of IPP Printer and Job objects via SNMP traps generated by the service provider (server or device).

2. Terminology

2.1. Terminology for Conformance

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted per [RFC-2119].

2.2. SNMP Network Management Framework

See: Section 1.1 'SNMPv1', section 1.2 'SNMPv2', and section 1.3 'SNMPv3' of [RFC-2576].

3. Model and Operation

3.1. SNMP Mapping for IPP Notifications

3.1.1. SNMP Mapping for IPP Printer Events

IPP Printer event attribute	Job Monitoring MIB object mapping
version-number	[no mapping - not useful]
status-code	[implicit in each generated trap]
notify-sequence-number	[request-id in SNMP trap header]
notify-charset	[no mapping - strings are UTF-8]
notify-natural-language	[no mapping - no text bindings]
subscription-id	[no mapping]
notify-subscribed-event	jmServiceEventNotifyTriggerEvent jmServiceEventNotifyGroupEvent
notify-printer-uri	jmServiceURI
printer-up-time	[sysUpTime in SNMP trap bindings]
printer-current-time	[hrSystemDate in Host Res MIB]
notify-user-data	[no mapping]
notify-text	[no mapping]
printer-state	jmServiceState
printer-state-reasons	jmServiceStateReasons
printer-is-accepting-jobs	[jmServiceStateReasons keyword]

3.1.2. SNMP Mapping for IPP Job Events

IPP Job event attribute	Job Monitoring MIB object mapping
version-number	[no mapping - not useful]
status-code	[implicit in each generated trap]
notify-sequence-number	[request-id in SNMP trap header]
notify-charset	[no mapping - strings are UTF-8]
notify-natural-language	[no mapping - no text bindings]
subscription-id	[no mapping]
notify-subscribed-event	jmJobEventNotifyTriggerEvent jmJobEventNotifyGroupEvent
notify-printer-uri	jmServiceURI
job-id	[via jmJobState OID instance]
printer-up-time	[sysUpTime in SNMP trap bindings]
printer-current-time	[hrSystemDate in Host Res MIB]
notify-user-data	[no mapping]
notify-text	[no mapping]
job-state	jmJobState
job-state-reasons	jmJobEventJobStateReasons [jmJobStateReasons1 plus zero or] [more jobStateReasonsN in a strin
[job-completed and job-purged - extra bindings]	
job-k-octets-processed	jmJobKOctetsProcessed
job-impressions-completed	jmJobImpressionsCompleted
[job-progress - extra bindings]	
job-k-octets	jmJobKOctetsPerCopyRequested
job-impressions	jmJobImpressionsPerCopyRequested
job-copies	jmProgressJobCopiesRequested
job-collation-type	jmProgressJobCollationType
job-media-sheets-completed	jmProgressMediaSheetsCompleted
sheet-completed-copy-number	jmProgressSheetCompletedCopyNum
sheet-completed-document-number	jmProgressSheetCompletedDocNum

3.2. Subscription for IPP Notifications

IPP Clients MAY subscribe for IPP Notifications delivered via SNMP by either of the following two standard methods:

3.2.1. Subscription via IPP

IPP Job creation (Create-Job, Print-Job, Print-URI) and Create-Subscription operations MAY be used to create per-Printer or per-Job IPP Subscription objects and MAY specify:

```
"notify-recipient-uri" = 'snmpnotify://hostname[.port]'

"notify-events" = <events of interest, e.g., 'job-completed'>

"notify-snmp-version" = <SNMP version and security model>

"notify-snmp-auth-data" = <authentication data, e.g., community>

"notify-snmp-operation" = <operation, e.g., SNMP Trap>

"notify-snmp-mtu-size" = <path max transmission unit, e.g., '4084'>
```

3.2.2. Subscription via SNMP

The SNMP Notification MIB and SNMP Target MIB (both in [RFC-2573]) MAY be used to create SNMP subscriptions.

To create a subscription via SNMP, the IPP Client MAY use an SNMP Set-Request operation to create an appropriate row in the SNMP Notification MIB with the following variable-bindings:

```
1)  'snmpNotifyTag'
    - tag of this notification - see 'snmpTargetAddrTagList' below
2)  'snmpNotifyType'
    - deliver via SNMP Trap (unacknowledged) or Inform (acknowledged)
3)  'snmpNotifyStorageType'
    - persistence (volatile, nonVolatile, permanent, readOnly)
```

Note: The SNMP Notification MIB uses persistence instead of the lease duration supported in IPP Subscription objects.

To complete a subscription via SNMP, the IPP Client MAY use an SNMP Set-Request operation to create an appropriate row in the SNMP Target MIB with the following variable-bindings:

1) 'snmpTargetAddrTDomain' and 'snmpTargetAddrTAddress'
   - notification target (client) transport protocol and address (corresponds to IPP 'notify-recipient-uri')
2) 'snmpTargetAddressTimeout' and 'snmpTargetAddrRetryCount'
   - retry timeout and limit (for acknowledged SNMP Inform only)
3) 'snmpTargetAddrTagList'
   - tags of notifications to be sent to this target (client)
4) 'snmpTargetAddrParamsEntry'
   - notification security and SNMP protocol version
5) 'snmpTargetAddrStorageType'
   - persistence (volatile, nonVolatile, permanent, readOnly)

Note: The SNMP Target MIB uses persistence instead of the lease duration supported in IPP Subscription objects.

3.3. Relationship to other MIBs

3.3.1. MIB-II (RFC 1213)

All SNMPv1 Trap messages include 'time-stamp' which is the value of the 'sysUpTime' object from MIB-II [RFC-1213].

All SNMPv2 trap bindings include the 'sysUpTime' object from MIB-II [RFC-1213].

3.3.2. Host Resources MIB (RFC 2790)

The 'jmServiceTable' defined in this document MAY have a sparse mapping to the 'hrDeviceTable' (same indices) in Host Resources MIB [RFC-2790].

3.3.3. Printer MIB (RFC 1759)

The 'jmServiceTable' defined in this document MAY have a sparse mapping to the 'hrDeviceTable' (same indices) in Host Resources MIB [RFC-2790], thus offering access to the Printer MIB via a 'hrDeviceIndex' value.

4. General Information

4.1. Table 1 - Information about the Delivery Method

Document Method Conformance Requirement	Delivery Method Realization
1. What is the URL scheme name for the Delivery Method?	snmpnotify
2. Is the Delivery Method REQUIRED, RECOMMENDED or OPTIONAL for an IPP Printer to support?	OPTIONAL
3. What transport and delivery protocols does the Printer use to deliver the Event Notification Content, i.e., what is the entire network stack?	SNMP application layer over UDP transport layer over IP network layer over any datalink layer (or SNMP over any other transport stack)
4. Can several Event Notifications be combined into a Compound Event Notification?	No, impossible in SNMP (in an interoperable fashion)
5. Is the Delivery Method initiated by the Notification Recipient (pull), or by the Printer (push)?	By the Printer (push)
6. Is the Event Notification content Machine Consumable or Human Consumable?	Machine Consumable
7. What section in this document answers the following question? For a Machine Consumable Event Notification, what is the representation and encoding of values defined in section 0 of [ipp-ntfy] and the conformance requirements thereof? For a Human Consumable Event Notification,	Section 7 'SNMP ASN.1 for for IPP Notifications'.

what is the representation and

encoding of pieces of information defined in section 9.2 of [ipp-ntfy] and the conformance requirements thereof?	
8. What are the latency and reliability of the transport and delivery protocol?	The latency is minimal in any network topology (all deployed transport mappings of SNMP are connectionless). The reliability is medium for SNMP Trap (unacknowledged) and high for SNMP Inform (acknowledged and optionally retried by the SNMP agent).
9. What are the security aspects of the transport and delivery protocol, e.g., how it is handled in firewalls?	SNMPv1 and SNMPv2 support community-based security with very weak authentication. SNMPv3 (RFC 257x series) supports user-based security with strong authentication and MAY traverse firewalls (but usually does not).
10. What are the content length restrictions?	SNMP messages MUST be encoded within the 'path MTU', i.e., the packet is NOT fragmented by the underlying network. In some IP networks, this results in a max payload of about 480 octets. See section 5.2.4 'notify-snmp-mtu-size'.
11. What are the additional values or pieces of information that a Printer sends in an Event Notification content and the conformance requirements thereof?	The 'job-completed' SNMP notification delivers 'job-k-octets-processed' as as well as the base required 'job-impressions-completed' - and both are REQUIRED (for interoperability)
12. What are the additional Subscription Template and/or Subscription Description attributes and the conformance requirements thereof?	See section 5.1 'Table 2 - Subscription and Printer Attributes' - all of the additional subscription template attributes are REQUIRED

(for interoperability)

13. What are the additional Printer Description attributes and the conformance requirements thereof?	See section 5.1 'Table 2 - Subscription and Printer Attributes' - all of the additional printer description attributes are REQUIRED (for interoperability).

5. Subscription Template Attributes

5.1. Table 2 - Subscription and Printer Attributes

All systems that support this specification MUST implement the following REQUIRED attributes defined in this specification (on the Subscription object or the Printer object):

Subscription Attribute Printer Attribute	Subscription Attribute Type Printer Attribute Type
notify-snmp-version	(type2 keyword)
notify-snmp-version-default	(type2 keyword)
notify-snmp-version-supported	(1setOf type2 keyword)
notify-snmp-auth-data	(octetString(MAX))
notify-snmp-auth-data-default	(octetString(MAX))
notify-snmp-auth-data-supported	(boolean)
notify-snmp-operation	(type2 keyword)
notify-snmp-operation-default	(type2 keyword)
notify-snmp-operation-supported	(1setOf type2 keyword)
notify-snmp-mtu-size	(integer(0:MAX))
notify-snmp-mtu-size-default	(integer(0:MAX))
notify-snmp-mtu-size-supported	(rangeOfInteger(0:MAX))

Note: The Printer attribute 'notify-snmp-auth-data-supported' has the irregular type 'boolean'.

5.2. Additional Subscription Template Attributes

5.2.1. notify-snmp-version (type2 keyword)

REQUIRED - for IPP Printers to implement.

This attribute specifies the SNMP protocol version and security model (e.g., community-based security in SNMPv1c IETF Standard [RFC-1157]) to use for delivery of this IPP Notification.

Note: SNMP protocol versions and SNMP security models are NOT mutually independent - therefore they are combined in this single attribute.

The following standard keyword values are defined:

'snmpv1-community': The SNMPv1 protocol with community-based security model defined in IETF Standard [RFC-1157].

'snmpv1-party': The SNMPv1 protocol with party-based security model defined in IETF Historic [RFC-1352].

'snmpv2-party': The SNMPv2 protocol with party-based security model defined in IETF Historic [RFC-1446].

'snmpv2-community': The SNMPv2 protocol with community-based security model defined in IETF Experimental [RFC-1902].

'snmpv2-user': The SNMPv2 protocol with user-based security model defined in IETF Experimental [RFC-1910].

'snmpv3-user': The SNMPv3 protocol with user-based security model defined in IETF Draft Standard [RFC-2574].

5.2.1.1. notify-snmp-version-default

REQUIRED - for IPP Printers to implement.
notify-snmp-version-default (type2 keyword)

IPP Printer object attribute for IPP Notifications over SNMP.

5.2.1.2. notify-snmp-version-supported

REQUIRED - for IPP Printers to implement.
notify-snmp-version-supported (1setOf type2 keyword)

IPP Printer object attribute for IPP Notifications over SNMP.

5.2.2. notify-snmp-auth-data (octetString(MAX))

REQUIRED - for IPP Printers to implement.

This attribute specifies the SNMP protocol authentication data (e.g., a community name in SNMPv1c IETF Standard [RFC-1157]) to use for delivery of this IPP Notification. This data MAY be opaque (not human-readable).

Human-readable authentication data in this attribute (e.g., an SNMPv1c or SNMPv2c community name) MUST be encoded in the UTF-8 [RFC-2279] transform of [ISO-10646] for interoperability.

5.2.2.1. notify-snmp-auth-data-default

REQUIRED - for IPP Printers to implement.
notify-snmp-auth-data-default (octetString(MAX))

IPP Printer object attribute for IPP Notifications over SNMP.

5.2.2.2. notify-snmp-auth-data-supported

REQUIRED - for IPP Printers to implement.
notify-snmp-auth-data-supported (boolean)

IPP Printer object attribute for IPP Notifications over SNMP.

Note: The Printer attribute 'notify-snmp-auth-data-supported' has the irregular type 'boolean'.

5.2.3. notify-snmp-operation (type2 keyword)

REQUIRED - for IPP Printers to implement.

This attribute specifies the SNMP operation (e.g, SNMP Trap in SNMPv1c IETF Standard [RFC-1157]) to use for delivery of this IPP Notification.

The following standard keyword values are defined:

'trap': An SNMP Trap PDU encoded in the version specified by "notify-snmp-version" above.
Note: SNMP Trap is an unconfirmed operation (i.e., the notification recipient MUST NOT reply with an acknowledgement) and MUST NOT be retried by the notification generator.

'inform': An SNMP Inform PDU encoded in the version specified by "notify-snmp-version" above.
Note: SNMP Inform is a confirmed operation (i.e., the notification recipient MUST reply with an acknowledgement Inform-Response) and MAY be retried by the notification generator.

'report': An SNMP Report PDU encoded in the version specified by "notify-snmp-version" above
Warning: The use of SNMP Report MAY not be interoperable.

5.2.3.1. notify-snmp-operation-default

REQUIRED - for IPP Printers to implement.
notify-snmp-operation-default (type2 keyword)

IPP Printer object attribute for IPP Notifications over SNMP.

5.2.3.2. notify-snmp-operation-supported

REQUIRED - for IPP Printers to implement.
notify-snmp-operation-supported (1setOf type2 keyword)

IPP Printer object attribute for IPP Notifications over SNMP.

5.2.4. notify-snmp-mtu-size (integer(0:MAX))

REQUIRED - for IPP Printers to implement.

This attribute specifies the SNMP path maximum transmission unit (MTU) size in octets to use for delivery of this IPP Notification. The SNMP path MTU is defined as the size of the complete SNMP message envelope, header, and body - but NOT of the underlying transport, network, and datalink headers - such that NO fragmentation is required for the SNMP message at ANY lower layer of the protocol stack OR on any segment of the bridged or routed delivery path (optimally).

This attribute enhances the integrity and probability of SNMP delivery.

This attribute may signal the need for algorithmic truncation of some string variable bindings in the SNMP notification which MAY not fit in very small path MTU sizes.

See: 'ifMtu' in the Interfaces group of IETF MIB-II [RFC-1213].

5.2.4.1. notify-snmp-mtu-size-default

REQUIRED - for IPP Printers to implement.
notify-snmp-mtu-size-default (integer(0:MAX))

IPP Printer object attribute for IPP Notifications over SNMP.

5.2.4.2. notify-snmp-mtu-size-supported

REQUIRED - for IPP Printers to implement.
notify-snmp-mtu-size-supported (rangeOfInteger(0:MAX))

IPP Printer object attribute for IPP Notifications over SNMP.

5.3. Additional Info about Subscription Template Attributes

5.3.1. notify-recipient-uri (uri)

This section describes the syntax of the value of this attribute for the 'snmpnotify' Delivery Method for IPP Notifications. The syntax for values of this attribute for other Delivery Methods for IPP Notifications is defined in each method's specification.

In order to support the 'snmpnotify' Delivery Method, the Printer MUST support the following syntax for the 'snmpnotify' Delivery Method when the Printer uses SNMP to deliver IPP Notifications. The lines below use RFC 2396 syntax rules, terms, and ABNF productions:

```
"snmpnotify://" hostport
hostport      = host [ ":" port ]
host          = hostname | IPv4address
hostname      = *( domainlabel "." ) toplabel [ "." ]
domainlabel   = alphanum | alphanum *( alphanum | "-" ) alphanum
toplabel      = alpha | alpha *( alphanum | "-" ) alphanum
IPv4address   = 1*digit "." 1*digit "." 1*digit "." 1*digit
port          = *digit

alphanum      = alpha | digit
alpha         = lowalpha | upalpha

lowalpha = "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" |
           "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" |
           "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z"
upalpha  = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" |
           "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" |
           "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"
digit    = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" |
           "8" | "9"
```

6. Event Notification Content

6.1. SNMP Message Envelope

6.1.1. SNMPv1 Message Envelope

The SNMPv1 [RFC-1157] message envelope for SNMPv1c (community-based security) contains the following:

6.1.1.1. version (INTEGER)

The SNMP protocol version (not the IPP version) for this message.

6.1.1.2. community (OCTET STRING)

The SNMP community name (read, write, or notify) for this message. Used for weak authentication.

6.1.2. SNMPv2 Message Envelope

The SNMPv2 [RFC-1905] message envelope for SNMPv2c (community-based security) [RFC-1901] contains the following:

6.1.2.1. version (INTEGER)

The SNMP protocol version (not the IPP version) for this message.

6.1.2.2. community (OCTET STRING)

The SNMP community name (read, write, or notify) for this message. Used for weak authentication.

6.1.3. SNMPv3 Message Envelope

The SNMPv2 [RFC-1905] message envelope for SNMPv3u (user-based security) [RFC-2574] contains the following:

6.1.3.1. msgVersion (INTEGER)

The SNMP protocol version (not the IPP version) for this message.

6.1.3.2. msgGlobalData (HeaderData)

Administrative parameters. The SNMP ASN.1 definition is as follows:

```
HeaderData ::= SEQUENCE {
    msgID      INTEGER (0..2147483647),
    msgMaxSize INTEGER (484..2147483647),
    msgFlags   OCTET STRING (SIZE(1)),
    msgSecurityModel INTEGER (1..2147483647)
}
```

6.1.3.3. msgSecurityParameters (OCTET STRING)

Security model-specific parameters.

See: 'User-based Security Model (USM) for SNMPv3' [RFC-2574].

6.1.3.4. msgData (ScopedPduData)

Scoped message body. The SNMP ASN.1 definition is as follows:

```
ScopedPduData ::= CHOICE {
    plaintext    ScopedPDU,
    encryptedPDU OCTET STRING  -- encrypted scopedPDU value
}

ScopedPDU ::= SEQUENCE {
    contextEngineID  OCTET STRING,
    contextName      OCTET STRING,
    data             ANY -- e.g., PDUs as defined in RFC 1905
}
```

6.2. SNMP Message Header

6.2.1. SNMPv1 Trap Message Header

Note: The 'request-id' (SNMP sequence number) is NOT encoded in the SNMPv1 Trap, whereas the SNMPv2 and SNMPv3 Trap/Inform requests DO

have 'request-id' in their message header.

6.2.1.1. enterprise (OBJECT IDENTIFIER)

The value of 'sysObjectID' (system type) in IETF MIB-II [RFC-1213] for the system that generated this SNMPv1 Trap.

6.2.1.2. agent-addr (NetworkAddress)

The network address of the SNMP agent that generated this SNMPv1 Trap.

6.2.1.3. generic-trap (INTEGER)

The base type of this SNMPv1 Trap: always 'enterpriseSpecific' in IPP Notifications).

6.2.1.4. specific-trap (INTEGER)

The specific type of this SNMPv1 Trap.

6.2.1.5. time-stamp (TimeTicks)

The value of 'sysUpTime' in IETF MIB-II [RFC-1213] when this SNMPv1 Trap was generated.

6.2.2. SNMPv2 and SNMPv3 Trap/Inform Message Header

Note: The 'request-id' (SNMP sequence number) is NOT encoded in the SNMPv1 Trap, whereas the SNMPv2 and SNMPv3 Trap/Inform requests DO have 'request-id' in their message header.

6.2.2.1. request-id (Integer32)

The receiver-specific sequence number for this SNMP message.

See: Section 5.4.2 'notify-sequence-number' in [IPP-NOT].

6.2.2.2. error-status (INTEGER)

The error status associated with this SNMP message.

6.2.2.3. error-index (INTEGER)

Unused in SNMP Trap messages.

6.3. SNMP Message Body

6.3.1. SNMPv1 Trap Message Body

The SNMPv1 Trap message body consists wholly of the specified parameters

See: Section 7.1 'Notification Definitions'. in the TRAP-TYPE definition.

6.3.2. SNMPv2 and SNMPv3 Trap/Inform Message Body

The SNMPv2 and SNMPv3 Trap/Inform message body consists of the specified parameters in the NOTIFICATION-TYPE definition PRECEDED by 'sysUpTime' from IETF MIB-II [RFC-1213] and 'sysTrapOID'.

See: Section 7.1 'Notification Definitions'.

6.3.2.1. sysUpTime (TimeTicks)

The value of 'sysUpTime' in IETF MIB-II [RFC-1213] when this SNMPv2 or SNMPv3 Trap/Inform was generated.

6.3.2.2. sysTrapOID (OBJECT IDENTIFIER)

The specific type of this SNMPv2 or SNMPv3 Trap/Inform. The value of the assignment statement at the end of the NOTIFICATION-TYPE definition.

6.4. SNMP Trap/Inform Examples

[To be supplied]

7. SNMP ASN.1 for IPP Notifications

7.1. Notification Definitions

7.1.1. Service Event Notify Group - Trap

```
--  Service Event Notify Group (CONDITIONALLY MANDATORY)
--
--  Implementation of this group is conditionally mandatory;
--  mandatory for systems which send Service traps via SNMP.

--  The jmServiceEventNotifyGroup consists entirely of the
--  jmServiceEventV2Notify notification.

jmServiceEventV1Enterprise OBJECT-IDENTITY
    STATUS      current
    DESCRIPTION
        "The value of the enterprise-specific OID in an SNMPv1 trap
        for a Service basic event sent by this managed system."
    ::= { jobmonMIBNotifications 1 }

jmServiceEventV2NotifyPrefix
    OBJECT IDENTIFIER ::= { jmServiceEventV1Enterprise 0 }

jmServiceEventV2Notify NOTIFICATION-TYPE
    OBJECTS {
        jmServiceEventNotifyTriggerEvent,
        jmServiceEventNotifyGroupEvent,
        jmServiceState,
        jmServiceStateReasons
    }
    STATUS      current
    DESCRIPTION
        "This SMIv2 trap corresponds to an IPP Printer basic event.

        The value of 'jmServiceIndex' for
        use with 'jmServiceTable' for this Service is conveyed in the
        instance qualifier (OID suffix) of 'jmServiceState'.

        This trap is sent when requested by a prior subscription.
        The specific event is in 'jmServiceEventNotifyTriggerEvent'.
        The group event is in 'jmServiceEventNotifyGroupEvent'.

        Additional variable-bindings SHOULD be appended to this trap:
```

- Systems with the Host Resources MIB (RFC 2790) SHOULD add

```
          'hrSystemDate'
          (compare to IPP 'printer-current-time')

        Additional variable-bindings MAY be appended to this trap
        for all printer-specific events:
        - Systems with the Host Resources MIB (RFC 2790) MAY add
          'hrDeviceStatus'
          (compare to IPP 'printer-state')
          'hrPrinterStatus'
          (compare to IPP 'printer-state') and
          'hrPrinterDetectedErrorState'
          (compare to IPP 'printer-state-reasons')

        Systems MAY add other variable-bindings from any MIB.

        See:    Standard events in 'jmServiceEventNotifyTriggerEvent';
                Section 5.3.2 'notify-events' in [IPP-NOT];
                Section 8.1 'notify-subscribed-event' in [IPP-NOT];
                Section 9 'Event Notification Content' in [IPP-NOT]."
    ::= { jmServiceEventV2NotifyPrefix 1 }
```

7.1.2. Job Event Notify Group - Trap

```
--  Job Event Notify Group (CONDITIONALLY MANDATORY)
--
--  Implementation of this group is conditionally mandatory;
--  mandatory for systems which send this Job trap via SNMP.

--  The jmJobEventNotifyGroup consists entirely of the
--  jmJobEventV2Notify notification.

jmJobEventV1Enterprise OBJECT-IDENTITY
    STATUS      current
    DESCRIPTION
        "The value of the enterprise-specific OID in an SNMPv1 trap
        for a Job basic event sent by this managed system."
    ::= { jobmonMIBNotifications 2 }

jmJobEventV2NotifyPrefix
    OBJECT IDENTIFIER ::= { jmJobEventV1Enterprise 0 }

jmJobEventV2Notify NOTIFICATION-TYPE
    OBJECTS {
        jmJobEventNotifyTriggerEvent,
        jmJobEventNotifyGroupEvent,
        jmJobState,
        jmJobEventJobStateReasons
    }
```

```
    STATUS      current
```

```
    DESCRIPTION
        "This SMIv2 trap corresponds to an IPP Job basic event.

        The values of 'jmGeneralJobSetIndex' and 'jmJobIndex' for
        use with 'jmJobTable' for this Job are conveyed in the
        instance qualifier (OID suffix) of 'jmJobState'.

        This trap is sent when requested by a prior subscription.
        The specific event is in 'jmJobEventNotifyTriggerEvent'.
        The group event is in 'jmJobEventNotifyGroupEvent'.

        Additional variable-bindings SHOULD be appended to this trap:
        - Systems with the Host Resources MIB (RFC 2790) SHOULD add
          'hrSystemDate'
          (compare to IPP 'printer-current-time')

        Systems MAY add other variable-bindings from any MIB.

        See:    Standard events in 'jmJobEventNotifyTriggerEvent';
                Section 5.3.2 'notify-events' in [IPP-NOT];
                Section 8.1 'notify-subscribed-event' in [IPP-NOT];
                Section 9 'Event Notification Content' in [IPP-NOT]."
    ::= { jmJobEventV2NotifyPrefix 1 }
```

7.1.3. Job Completed Notify Group - Trap

```
--  Job Completed Notify Group (CONDITIONALLY MANDATORY)
--
--  Implementation of this group is conditionally mandatory;
--  mandatory for systems which send this Job trap via SNMP.

jmJobCompletedV1Enterprise OBJECT-IDENTITY
    STATUS      current
    DESCRIPTION
        "The value of the enterprise-specific OID in an SNMPv1 trap
        for a Job completed event sent by this managed system."
    ::= { jobmonMIBNotifications 3 }

jmJobCompletedV2NotifyPrefix
    OBJECT IDENTIFIER ::= { jmJobCompletedV1Enterprise 0 }

jmJobCompletedV2Notify NOTIFICATION-TYPE
    OBJECTS {
        jmJobState,
        jmJobEventJobStateReasons,
        jmJobKOctetsProcessed,
        jmJobImpressionsCompleted
    }
```

```
    STATUS      current
```

```
    DESCRIPTION
        "This SMIv2 trap corresponds to an IPP 'job-completed' event.

        The values of 'jmGeneralJobSetIndex' and 'jmJobIndex' for
        use with 'jmJobTable' for this Job are conveyed in the
        instance qualifier (OID suffix) of 'jmJobState'.

        This trap is sent when requested by a prior subscription.
        The event type is 'job-completed'.

        Additional variable-bindings SHOULD be appended to this trap:
        - Systems with the Host Resources MIB (RFC 2790) SHOULD add
          'hrSystemDate'
          (compare to IPP 'printer-current-time')

        Systems MAY add other variable-bindings from any MIB.

        See:    Section 5.3.2 'notify-events' in [IPP-NOT];
                Section 8.1 'notify-subscribed-event' in [IPP-NOT];
                Section 9 'Event Notification Content' in [IPP-NOT]."
    ::= { jmJobCompletedV2NotifyPrefix 1 }
```

7.1.4. Job Progress Notify Group - Trap

```
--  Job Progress Notify Group (CONDITIONALLY MANDATORY)
--
--  Implementation of this group is conditionally mandatory;
--  mandatory for systems which send this Job trap via SNMP.

jmJobProgressV1Enterprise OBJECT-IDENTITY
    STATUS      current
    DESCRIPTION
        "The value of the enterprise-specific OID in an SNMPv1 trap
        for a Job progress event sent by this managed system."
    ::= { jobmonMIBNotifications 4 }

jmJobProgressV2NotifyPrefix
    OBJECT IDENTIFIER ::= { jmJobProgressV1Enterprise 0 }

jmJobProgressV2Notify NOTIFICATION-TYPE
    OBJECTS {
        jmJobKOctetsPerCopyRequested,
        jmJobKOctetsProcessed,
        jmJobImpressionsPerCopyRequested,
        jmJobImpressionsCompleted,
        jmProgressJobCopiesRequested,
        jmProgressJobCollationType,
        jmProgressMediaSheetsCompleted,
```

jmProgressSheetCompletedCopyNum,

```
        jmProgressSheetCompletedDocNum
    }
    STATUS      current
    DESCRIPTION
        "This SMIv2 trap corresponds to an IPP 'job-progress' event.

        The values of 'jmGeneralJobSetIndex' and 'jmJobIndex' for
        use with 'jmJobTable' for this Job are conveyed in the
        instance qualifier (OID suffix) of 'jmJobKOctetsProcessed'.

        This trap is sent when requested by a prior subscription.
        The event type is 'job-progress'.

        Additional variable-bindings SHOULD be appended to this trap:
        - Systems with the Host Resources MIB (RFC 2790) SHOULD add
          'hrSystemDate'
          (compare to IPP 'printer-current-time')

        Systems MAY add other variable-bindings from any MIB.

        See:    Section 5.3.2 'notify-events' in [IPP-NOT];
                Section 8.1 'notify-subscribed-event' in [IPP-NOT];
                Section 9 'Event Notification Content' in [IPP-NOT]."
    ::= { jmJobProgressV2NotifyPrefix 1 }
```

7.2. Object Definitions

7.2.1. Service Group - Objects

```
--  The Service Group (CONDITIONALLY MANDATORY)
--
--  Implementation of this group is conditionally mandatory;
--  mandatory for systems which show Service state via SNMP.

--  The jmServiceGroup consists entirely of the jmServiceTable

jmService  OBJECT IDENTIFIER ::= { jobmonMIBObjects 7 }

jmServiceTable  OBJECT-TYPE
    SYNTAX      SEQUENCE OF JmServiceEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The jmServiceTable consists of basic service state and status
        information for each service which offers one or more job
        services on this managed system.
```

```
        An entry SHALL exist in this table for each service, no matter
        what the state of that service.  A service MAY support multiple
        configured job sets and configured devices.

        See:    'jmServiceJobSetsConfigured' and
                'jmServiceDevicesConfigured' bit-arrays in this MIB."
    ::= { jmService 1 }


jmServiceEntry  OBJECT-TYPE
    SYNTAX      JmServiceEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Basic service state and status information."
    INDEX  { jmServiceIndex }
    ::= { jmServiceTable 1 }


JmServiceEntry ::= SEQUENCE {
    jmServiceIndex                      Integer32 (1..2147483647),
    jmServiceName                       JmUTF8StringTC (SIZE (0..63)),
    jmServiceURI                        JmUTF8StringTC (SIZE (0..63)),
    jmServiceJobServiceTypes            JmJobServiceTypesTC,
    jmServiceJobSetsConfigured          OCTET STRING (SIZE (0..255)),
    jmServiceDevicesConfigured          OCTET STRING (SIZE (0..255)),
    jmServiceState                      JmServiceStateTC,
    jmServiceStateReasons               SnmpAdminString (SIZE (0..255))
}

jmServiceIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The unique identifier for this service on this managed system.

        See:    'jmServiceEventServiceIndex' object in this MIB."
    ::= { jmServiceEntry 1 }

jmServiceName OBJECT-TYPE
    SYNTAX      JmUTF8StringTC (SIZE(0..63))    -- 127 in IPP
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The human readable name of this managed service.

        See:    'deviceNameRequested' job attribute in this MIB;
                'physicalDevice' job attribute in this MIB;
```

Section 4.4.4 'printer-name' in [IPP-MOD]."

```
    DEFVAL      { ''H }                 -- no service name
    ::= { jmServiceEntry 2 }

jmServiceURI OBJECT-TYPE
    SYNTAX      JmUTF8StringTC (SIZE(0..63))    -- 1023 in IPP
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "A URI for this managed service (valid for job services).

        See:    Section 4.3.3 'job-printer-uri' in [IPP-MOD];
                Section 4.4.1 'printer-uri-supported' in [IPP-MOD]."
    DEFVAL      { ''H }                 -- no service URI
    ::= { jmServiceEntry 3 }

jmServiceJobServiceTypes OBJECT-TYPE
    SYNTAX      JmJobServiceTypesTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The types of job services supported by this managed service.

        See:    'JmJobServiceTypesTC' textual convention in this MIB;
                'jobServiceTypes' job attribute in this MIB."
    DEFVAL      { 0 }                   -- no job services
    ::= { jmServiceEntry 4 }

jmServiceJobSetsConfigured OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE(0..255))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "A bit-array that specifies the job sets configured for this
        service, where each bit '2**n' is set if 'jmGeneralJobSetIndex'
        is a configured job set.   Uses network byte order (big-endian)
        rules - the high-order bit of the first octet corresponds to
        'jmGeneralJobSetIndex' of '0' (reserved) - the low-order bit of
        the first octet corresponds to 'jmGeneralJobSetIndex' of '7'.
        Supports values of 'jmGeneralJobSetIndex' from '1' to '2039'.

        Compare to the BITS pseudotype defined in IETF SMIv2 (RFC 2578)
        which has the same bit ordering rules (big-endian).

        See:    'queueNameRequested' job attribute in this MIB;
                'jmGeneralJobSetIndex' table index in this MIB."
    DEFVAL      { ''H }                 -- no job sets configured
    ::= { jmServiceEntry 5 }

jmServiceDevicesConfigured OBJECT-TYPE
```

```
    SYNTAX      OCTET STRING (SIZE(0..255))
    MAX-ACCESS  read-only
```

```
    STATUS      current
    DESCRIPTION
        "A bit-array that specifies the devices configured for this
        service, where each bit '2**n' is set if 'hrDeviceIndex'
        is a configured device.   Uses network byte order (big-endian)
        rules - the high-order bit of the first octet corresponds to
        'hrDeviceIndex' of '0' (reserved) - the low-order bit of
        the first octet corresponds to 'hrDeviceIndex' of '7'.
        Supports values of 'hrDeviceIndex' from '1' to '2039'.

        Compare to the BITS pseudotype defined in IETF SMIv2 (RFC 2578)
        which has the same bit ordering rules (big-endian).

        See:    'physicalDevice' job attribute in this MIB;
                'hrDeviceIndex' in IETF Host MIB (RFC 2790)."
    DEFVAL      { ''H }                 -- no devices configured
    ::= { jmServiceEntry 6 }

jmServiceState OBJECT-TYPE
    SYNTAX      JmServiceStateTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current state of this managed service.

        See:    'jmServiceEventServiceState' object in this MIB;
                'jmJobState' object in this MIB;
                Section 4.4.11 'printer-state' in [IPP-MOD]."
    DEFVAL      { unknown }             -- unknown service state
    ::= { jmServiceEntry 7 }

jmServiceStateReasons OBJECT-TYPE
    SYNTAX      SnmpAdminString (SIZE (0..255)) -- multi-valued in IPP
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The service state reasons associated with this service state
        (as a comma-separated list) or the empty string (if none).

        See:    'jmServiceEventServiceStateReasons' object in this MIB;
                'jmJobStateReasons1' object in this MIB;
                Section 4.4.12 'printer-state-reasons' in [IPP-MOD]."
    DEFVAL      { ''H }                 -- no service state reasons
    ::= { jmServiceEntry 8 }
```

7.2.2. Service Event Group - Objects

```
--  The Service Event Group (CONDITIONALLY MANDATORY)
```

--

```
--  Implementation of this group is conditionally mandatory;
--  mandatory for systems which show Service events via SNMP.

--  The jmServiceEventGroup consists entirely of the jmServiceEventTable

jmServiceEvent  OBJECT IDENTIFIER ::= { jobmonMIBObjects 8 }

jmServiceEventTable  OBJECT-TYPE
    SYNTAX      SEQUENCE OF JmServiceEventEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The jmServiceEventTable contains service event entries for the
        services which offer job services on this managed system."
    ::= { jmServiceEvent 1 }


jmServiceEventEntry  OBJECT-TYPE
    SYNTAX      JmServiceEventEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Basic service event information."
    INDEX  { jmServiceEventIndex }
    ::= { jmServiceEventTable 1 }


JmServiceEventEntry ::= SEQUENCE {
    jmServiceEventIndex                 Integer32 (1..2147483647),
    jmServiceEventNotifyTriggerEvent    SnmpAdminString (SIZE (0..63)),
    jmServiceEventNotifyGroupEvent      SnmpAdminString (SIZE (0..63)),
    jmServiceEventNotifyTime            TimeTicks,
    jmServiceEventServiceIndex          Integer32 (1..2147483647),
    jmServiceEventServiceState          JmServiceStateTC,
    jmServiceEventServiceStateReasons   SnmpAdminString (SIZE (0..255))
}

jmServiceEventIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The unique identifier for this event on this managed system."
    ::= { jmServiceEventEntry 1 }

jmServiceEventNotifyTriggerEvent OBJECT-TYPE
    SYNTAX      SnmpAdminString (SIZE (0..63))   -- 255 in [IPP-NOT]
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
```

"Specific event that created this row in 'jmServiceEventTable'.

```
        For example, 'printer-stopped' here and 'printer-state-changed'
        in 'jmServiceEventNotifyGroupEvent'.  Whereas a transition of
        'jmServiceState' from 'idle' to 'processing' would report
        'printer-state-changed' in both 'jmService...Event' objects.

        Standard Printer event types defined in [IPP-NOT] are:
        - 'printer-state-changed' (group event)
          - 'printer-restarted'
          - 'printer-shutdown'
          - 'printer-stopped'
        - 'printer-config-changed' (group event)
          - 'printer-media-changed'
          - 'printer-finishings-changed'
        - 'printer-queue-order-changed' (group event)

        Standard Service event types generalized from [IPP-NOT] are:
        - 'service-state-changed' (group event)
          - 'service-restarted'
          - 'service-shutdown'
          - 'service-stopped'
        - 'service-config-changed' (group event)
          - 'service-media-changed'
          - 'service-finishings-changed'
        - 'service-queue-order-changed' (group event)
        - and (optionally) vendor extension event types

        Conformance:    The natural language for keywords
        in event type SHALL always be US English.

        See:    'jmServiceEventNotifyGroupEvent' in this MIB;
                Section 8.1 'notify-subscribed-event' in [IPP-NOT]."
    DEFVAL      { ''H }                 -- no notify event
    ::= { jmServiceEventEntry 2 }

jmServiceEventNotifyGroupEvent OBJECT-TYPE
    SYNTAX      SnmpAdminString (SIZE (0..63))   -- 255 in [IPP-NOT]
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Group event that created this row in 'jmServiceEventTable'.
        For example, 'printer-state-changed' here and 'printer-stopped'
        in 'jmServiceEventNotifyTriggerEvent'.  Whereas a transition of
        'jmServiceState' from 'idle' to 'processing' would report
        'printer-state-changed' in both 'jmService...Event' objects.

        Conformance:    The natural language for keywords
        in event type SHALL always be US English.

        See:    Standard events in 'jmServiceEventNotifyTriggerEvent';
```

```
                Section 8.1 'notify-subscribed-event' in [IPP-NOT]."
    DEFVAL      { ''H }                 -- no notify event
```

```
    ::= { jmServiceEventEntry 3 }

jmServiceEventNotifyTime OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The time of this service event.

        Usage:  Conforming management agents, which MUST implement the
        IETF MIB-II (RFC 1213), SHALL set 'jmServiceEventNotifyTime' to
        'sysUpTime' when a service event row is created.

        See:    'sysUpTime' in IETF MIB-II (RFC 1213);
                Section 5.4.4 'notify-printer-up-time' in [IPP-NOT];
                Section 4.4.29 'printer-up-time' in [IPP-MOD]."
    ::= { jmServiceEventEntry 4 }

jmServiceEventServiceIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The unique identifier for this service on this managed system.

        See:    'jmServiceIndex' object in this MIB."
    ::= { jmServiceEventEntry 5 }

jmServiceEventServiceState OBJECT-TYPE
    SYNTAX      JmServiceStateTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The state of this managed service at the time of this event.

        See:    'jmServiceState' object in this MIB;
                'jmJobState' object in this MIB;
                Section 4.4.11 'printer-state' in [IPP-MOD]."
    DEFVAL      { unknown }             -- unknown service state
    ::= { jmServiceEventEntry 6 }

jmServiceEventServiceStateReasons OBJECT-TYPE
    SYNTAX      SnmpAdminString (SIZE (0..255)) -- multi-valued in IPP
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The service state reasons associated with this service state
        (as a comma-separated list) or the empty string (if none).

        See:    'jmServiceStateReasons' object in this MIB;
```

'jmJobStateReasons1' object in this MIB;

```
                Section 4.4.12 'printer-state-reasons' in [IPP-MOD]."
    DEFVAL      { ''H }                 -- no service state reasons
    ::= { jmServiceEventEntry 7 }
```

7.2.3. Job Event Group - Objects

```
--  The Job Event Group (CONDITIONALLY MANDATORY)
--
--  Implementation of this group is conditionally mandatory;
--  mandatory for systems which show Job events via SNMP.

--  The jmJobEventGroup consists entirely of the jmJobEventTable

jmJobEvent  OBJECT IDENTIFIER ::= { jobmonMIBObjects 9 }

jmJobEventTable  OBJECT-TYPE
    SYNTAX      SEQUENCE OF JmJobEventEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The jmJobEventTable contains job event entries for the
        jobs present on this managed system."
    ::= { jmJobEvent 1 }


jmJobEventEntry  OBJECT-TYPE
    SYNTAX      JmJobEventEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Basic job event information."
    INDEX  { jmJobEventIndex }
    ::= { jmJobEventTable 1 }


JmJobEventEntry ::= SEQUENCE {
    jmJobEventIndex                     Integer32 (1..2147483647),
    jmJobEventNotifyTriggerEvent        SnmpAdminString (SIZE (0..63)),
    jmJobEventNotifyGroupEvent          SnmpAdminString (SIZE (0..63)),
    jmJobEventNotifyTime                TimeTicks,
    jmJobEventJobSetIndex               Integer32 (1..32767),
    jmJobEventJobIndex                  Integer32 (1..2147483647),
    jmJobEventJobState                  JmJobStateTC,
    jmJobEventJobStateReasons           OCTET STRING (SIZE (4..16))
}

jmJobEventIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
```

```
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
    DESCRIPTION
        "The unique identifier for this event on this managed system."
    ::= { jmJobEventEntry 1 }

jmJobEventNotifyTriggerEvent OBJECT-TYPE
    SYNTAX      SnmpAdminString (SIZE (0..63))   -- 255 in [IPP-NOT]
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specific event that created this row in 'jmJobEventTable'.
        For example, 'job-stopped' here and 'job-state-changed'
        in 'jmJobEventNotifyGroupEvent'.  Whereas a transition of
        'jmJobState' from 'pending' to 'processing' would report
        'job-state-changed' in both 'jmJob...Event' objects.

        Standard Job event types defined in [IPP-NOT] are:
        - 'job-state-changed' (group event)
          - 'job-created'
          - 'job-completed'
          - 'job-stopped'
        - 'job-config-changed' (group event)
        - 'job-progress' (group event)
        - and (optionally) vendor extension event types

        Conformance:    The natural language for keywords
        in event type SHALL always be US English.

        See:    'jmJobEventNotifyGroupEvent' in this MIB;
                Section 8.1 'notify-subscribed-event' in [IPP-NOT]."
    DEFVAL      { ''H }                 -- no notify event
    ::= { jmJobEventEntry 2 }

jmJobEventNotifyGroupEvent OBJECT-TYPE
    SYNTAX      SnmpAdminString (SIZE (0..63))   -- 255 in [IPP-NOT]
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Group event that created this row in 'jmJobEventTable'.
        For example, 'job-state-changed' here and 'job-stopped'
        in 'jmJobEventNotifyTriggerEvent'.  Whereas a transition of
        'jmJobState' from 'pending' to 'processing' would report
        'job-state-changed' in both 'jmJob...Event' objects.

        Conformance:    The natural language for keywords
        in event type SHALL always be US English.

        See:    'jmJobEventNotifyGroupEvent' in this MIB;
                Section 8.1 'notify-subscribed-event' in [IPP-NOT]."
```

```
    DEFVAL      { ''H }                 -- no notify event
    ::= { jmJobEventEntry 3 }
```

```
jmJobEventNotifyTime OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The time of this job event.

        Usage:  Conforming management agents, which MUST implement the
        IETF MIB-II (RFC 1213), SHALL set 'jmJobEventNotifyTime' to
        'sysUpTime' when a job event row is created.

        See:    'sysUpTime' in IETF MIB-II (RFC 1213);
                Section 5.4.4 'notify-printer-up-time' in [IPP-NOT];
                Section 4.4.29 'printer-up-time' in [IPP-MOD]."
    ::= { jmJobEventEntry 4 }

jmJobEventJobSetIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..32767)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The unique identifier for this job set on this managed system.

        See:    'jmGeneralJobSetIndex' object in this MIB."
    ::= { jmJobEventEntry 5 }

jmJobEventJobIndex OBJECT-TYPE
    SYNTAX      Integer32 (1..2147483647)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The unique identifier for this job on this managed system,
        when prefixed with 'jmJobEventJobSetIndex'.

        See:    'jmJobIndex' object in this MIB."
    ::= { jmJobEventEntry 6 }

jmJobEventJobState OBJECT-TYPE
    SYNTAX      JmJobStateTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The state of this managed job at the time of this event.

        See:    'jmJobState' in this MIB;
                Section 4.3.7 'job-state' in [IPP-MOD]."
    DEFVAL      { unknown }             -- unknown job state
    ::= { jmJobEventEntry 7 }
```

jmJobEventJobStateReasons OBJECT-TYPE

```
    SYNTAX      OCTET STRING (SIZE (4..16)) -- multi-valued in IPP
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The job state reasons associated with this job state
        represented as one to four concatenated 32-bit integers
        in network byte order (big-endian).

        Usage:  Conforming management stations SHALL always report the
        value of the object 'jmJobStateReasons1' in the first four
        octets of 'jmJobEventJobStateReasons' and SHOULD report values
        of the attributes 'jobStateReasons2', 'jobStateReasons3', and
        'jobStateReasons4' in subsequent octets.

        See:    'jmJobStateReasons1' in this MIB;
                Section 4.3.8 'job-state-reasons' in [IPP-MOD]."
    DEFVAL      { '00000000'H }         -- no job state reasons
    ::= { jmJobEventEntry 8 }
```

7.2.4. Job Progress Group - Objects

```
--  The Job Progress Group (CONDITIONALLY MANDATORY)
--
--  Implementation of this group is conditionally mandatory;
--  mandatory for systems which send Job progress traps via SNMP.

--  The jmProgressGroup consists entirely of leaf objects for traps

jmProgress  OBJECT IDENTIFIER ::= { jobmonMIBObjects 10 }

jmProgressJobCopiesRequested OBJECT-TYPE
    SYNTAX      Integer32 (-2..2147483647)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of copies of this job requested by the client.

        See:    'jobCopiesRequested' attribute in this MIB."
    DEFVAL      { -2 }                  -- unknown job copies
    ::= { jmProgress 1 }

jmProgressJobCollationType OBJECT-TYPE
    SYNTAX      JmJobCollationTypeTC
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of copies of this job requested by the client.
```

See: 'jobCollationType' attribute in this MIB;

```
                'job-collation-type' in [IPP-PROG];
                Section 9 'Event Notification Content' in [IPP-NOT]."
    DEFVAL      { unknown }             -- unknown job collation type
    ::= { jmProgress 2 }

jmProgressMediaSheetsCompleted OBJECT-TYPE
    SYNTAX      Integer32 (-2..2147483647)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of media sheets completed for this job.

        See:    'sheetsCompleted' attribute in this MIB;
                Section 4.3.18.3 'job-media-sheets-completed'
                in [IPP-MOD];
                Section 9 'Event Notification Content' in [IPP-NOT]."
    DEFVAL      { -2 }                  -- unknown job progress
    ::= { jmProgress 3 }

jmProgressSheetCompletedCopyNum OBJECT-TYPE
    SYNTAX      Integer32 (-2..2147483647)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of the job copy currently being stacked for the
        current document or zero (if none) or '-2' (unknown).

        See:    'sheetCompletedCopyNumber' attribute in this MIB;
                'sheet-completed-copy-number' in [IPP-PROG];
                Section 9 'Event Notification Content' in [IPP-NOT]."
    DEFVAL      { -2 }                  -- unknown sheet complete copy
    ::= { jmProgress 4 }

jmProgressSheetCompletedDocNum OBJECT-TYPE
    SYNTAX      Integer32 (-2..2147483647)
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of the job document currently being stacked for the
        current job or zero (if none) or '-2' (unknown).

        See:    'sheetCompletedDocumentNumber' attribute in this MIB;
                'sheet-completed-document-number' in [IPP-PROG];
                Section 9 'Event Notification Content' in [IPP-NOT]."
    DEFVAL      { -2 }                  -- unknown sheet complete doc
    ::= { jmProgress 5 }
```

8. Conformance

See DESCRIPTION clauses of objects and notifications defined in section 7 **'SNMP ASN.1 for IPP Notifications'.**

9. IANA Considerations

None.

10. Internationalization Considerations

The 'jmServiceStateReasons', 'jmServiceEventServiceStateReasons', 'jmServiceEventNotify[Trigger|Group]Event', and 'jmJobEventNotify[Trigger|Group]Event' text string objects defined in this document have a syntax (datatype) of 'SnmpAdminString' (UTF-8 stream-encoded ISO 10646/Unicode text) defined in the SNMPv3 Framework MIB [RFC-2571]. The natural language of these objects is US English (they contain keywords which are protocol elements).

11. Security Considerations

This IPP Notifications over SNMP mapping defines only 'read-only' objects. It is suitable for use with any version of SNMP, as no update security is required (because no configuration updates are supported).

No sensitive information is available via IPP Notifications over SNMP.

12. References

[IPP-MOD] R. deBry, T. Hastings, R. Herriot, S. Isaacson, P. Powell. IPP/1.1 Model and Semantics, <draft-ietf-ipp-model-v11-07.txt> (work-in-progress), May 2000.

[IPP-NOT] S. Isaacson, J. Martin, R. deBry, T. Hastings, M. Shepherd, R. Bergman. IPP Event Notification Specification, <draft-ietf-ipp-not-spec-04.txt> (work-in-progress), July 2000.

[IPP-PROG] T. Hastings, R. Bergman, H. Lewis. Proposed Job Progress Attributes for IPP, <draft-ietf-ipp-job-prog-00.txt> (work-in-progress), July 2000.

[RFC-1157] J.D. Case, M. Fedor, M.L. Schoffstall, C. Davin. Simple Network Management Protocol, RFC 1157, May 1990.

[RFC-1213] K. McCloghrie, M. Rose. MIB-II, RFC 1213, March 1991.

[RFC-1215] M. Rose. Convention for Defining Traps for SNMPv1, RFC 1215, March 1991.

[RFC-1759] R. Smith, F. Wright, T. Hastings, S. Zilles, J. Gyllenskog. Printer MIB, RFC 1759, March 1995.

[RFC-1901] J. Case, K. McCloghrie, M. Rose, S. Waldbusser. Introduction to Community-based SNMPv2, RFC 1901, January 1996.

[RFC-1905] J. Case, K. McCloghrie, M. Rose, S. Waldbusser. Protocol Operations for SNMPv2, RFC 1905, January 1996.

[RFC-2396] T. Berners-Lee, R. Fielding, L. Masinter. Uniform Resource Identifiers (URI): Generic Syntax, RFC 2396, August 1998.

[RFC-2571] D. Harrington, R. Presuhn, B. Wijnen. Architecture for SNMP Management Frameworks, RFC 2571, April 1999.

[RFC-2573] D. Levi, P. Meyer, B. Stewart. SNMP Applications, RFC 2573, April 1999.

[RFC-2574] U. Blumenthal, B. Wijnen. User-based Security Model (USM) for SNMPv3, RFC 2574, April 1999.

[RFC-2575] B. Wijnen, R. Presuhn, K. McCloghrie. View-based Access Control Model (VACM) for SNMPv3, RFC 2575, April 1999.

[RFC-2576] R. Frye, D. Levi, S. Routhier, B. Wijnen. Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard

Network Management Framework, RFC 2576, March 2000.

[RFC-2707] R. Bergman, T. Hastings, S. Isaacson, H. Lewis. Job Monitoring MIB v1.0, RFC 2707, November 1999

[RFC-2790] P. Grillo, S. Waldbusser. Host Resources MIB, RFC 2790, March 2000.

13. Change Log

[This section will be deleted before publication as an RFC]

Changes in reverse chronological order (most recent first).

<draft-ietf-ipp-not-over-snmp-04.txt> - 8 August 2000

1) Deleted 'printer-full', 'printer-no-longer-full', 'printer-almost-idle', and 'printer-not-almost-idle' event types, and added 'printer-stopped' in 'jmServiceEventNotifyTriggerEvent', for alignment with revised [IPP-NOT];
2) Deleted 'job-purged' event type and added 'job-stopped' in 'jmJobEventNotifyTriggerEvent', for alignment with revised [IPP-NOT];
3) Renamed 'jmServiceEventNotifyEvent' object to 'jmServiceEventNotifyTriggerEvent' (most specific event) and added 'jmServiceEventNotifyGroupEvent' (most general event), for higher fidelity mapping to IPP 'notify-subscribed-event';
4) Renamed 'jmJobEventNotifyEvent' object to 'jmJobEventNotifyTriggerEvent' (most specific event) and added 'jmJobEventNotifyGroupEvent' (most general event), for higher fidelity mapping to IPP 'notify-subscribed-event';
5) Renamed all NOTIFICATION-TYPEs (traps) for clarity, changing 'jmServiceBasicV2Event' to 'jmServiceEventV2Notify', 'jmJobBasicV2Event' to 'jmJobEventV2Notify', 'jmJobCompletedV2Event' to 'jmJobCompletedV2Notify', and 'jmJobProgressV2Event' to 'jmJobProgressV2Notify'.

<draft-ietf-ipp-not-over-snmp-03.txt> - 6 July 2000

1) Added 'SnmpAdminString' to IMPORTS clause for new objects.
2) Corrected OID in MODULE-IDENTITY to use forward reference to definition of 'pwg' from 'enterprises' and 'mibs' from 'pwg'.
3) Added 'JmServiceStateTC' textual convention.
4) Added 'jmMirrorAttr' and 'jmSystem' object identifiers reserved for future extensions.
5) Major rewrite, per email discussion on IETF IPP WG list, to specify four new small (traditional) SNMP traps for: 'jmServiceBasicV2Event' (generalized from IPP Printer event), 'jmJobBasicV2Event' (corresponds IPP Job event), 'jmJobCompletedV2Event' (corresponds IPP Job completed event), 'jmJobProgressV2Event' (corresponds IPP Job progress event).
6) Major rewrite, per email discussion on IETF IPP WG list, to specify four new SNMP object groups: 'jmServiceTable' (name, URI, state, etc. - from IPP Printer), 'jmServiceEventTable' (records IPP Printer events for polling), 'jmJobEventTable' (records IPP Job events for polling), 'jmJobProgressGroup' (leaf objects for IPP Job progress event).

7) Revised section 3.1 'SNMP Mapping for IPP Printer Events' and section 3.2 'SNMP Mapping for IPP Job Events', to agree with above.

8) Deleted obsolete section 3.3 'Rules for Encoding Notifications', as event bindings now always fit over all SNMP transport protocols.

<draft-ietf-ipp-not-over-snmp-02.txt> - 19 March 2000

1) Renamed Printer Event notification group to Device Basic Event and 'jmPrinterEventV2Event' notification to 'jmDeviceBasicV2Event', to better align with IPP and to support non-printing jobs.
2) Revised 'jmDeviceBasicV2Event' notification to remove 'hrDeviceStatus', 'hrPrinterStatus', 'hrPrinterDetectedErrorState' from mandatory trap bindings because they were redundant, per request of Ron Bergman.
3) Renamed Job Event notification group to Job Basic Event and 'jmJobEventV2Event' notification to 'jmJobBasicV2Event', to better align with IPP and to support variant job events.
4) Defined new Job Completed Event notification group and defined new Job Progress Event notification group, to better align with IPP and to support variant job events.
5) Renamed Event object group to Event Binding, 'jmEventPrinterState' to 'jmEventDeviceState', 'jmEventPrinterStateReasons' to 'jmEventDeviceStateReasons', 'jmEventPrinterIsAcceptingJobs' to 'jmEventDeviceIsAcceptingJobs', to support non-printing jobs.
6) Revised Event Binding object group, adding explicit objects 'jmEventDeviceURI', 'jmEventDeviceName', 'jmEventJobSetIndex', 'jmEventJobIndex', 'jmEventJobName', 'jmEventJobState', 'jmEventJobStateReasons', 'jmEventJobKOctets', 'jmEventJobKOctetsProcessed', 'jmEventJobImpressions', 'jmEventJobImpressionsCompleted', 'jmEventJobMediaSheets', 'jmEventJobMediaSheetsCompleted', 'jmEventJobImpressionsCompletedCC', 'jmEventJobCollationType', 'jmEventJobSheetCompletedCopyNum', 'jmEventJobSheetCompletedDocNum', per request of Ron Bergman.
7) Revised SYNTAX of 'jmEventTriggerEvent' object from from 'JmUTF8StringTC' (string) to IPP-aligned enumeration, per request of Ron Bergman.
8) Removed all references to Printer MIB v2, as they were of limited value, per request of Ron Bergman.
9) Revised 'SNMP Mapping for IPP Printer Events' section for renamed event binding objects, per request of Ron Bergman.
10) Revised 'Rules for Encoding Notifications' section to truncate additional string bindings, per request of Ron Bergman.
11) Revised 'Registration via IPP' section, to change scheme name from 'ipp-snmp:' to 'snmpnotify:', per request of Ron Bergman.

<draft-ietf-ipp-not-over-snmp-01.txt> - 1 December 1999

1) Deleted 'JmTriggerEventTC' textual convention (see below).
2) Revised SYNTAX of 'jmEventTriggerEvent' object from 'JmTriggerEventTC' (enumeration) to 'JmUTF8StringTC' (string), to support use of IPP standard keywords.

3) Added 'jmEventPrinterState', 'jmEventPrinterStateReasons', and 'jmEventPrinterIsAcceptingJobs' objects for consistency w/

[IPP-NOT] and to reduce ambiguity about printer states inherent in RFC 1759.

4) Revised DESCRIPTION of 'jmPrinterEventV2Event' notification to add SHOULD (recommendation) for 'jmEventPrinterState', 'jmEventPrinterStateReasons', and 'jmEventPrinterIsAcceptingJobs' objects.

5) Revised 'SNMP Mapping for IPP Printer Events' section to add direct mapping of IPP notification attributes to 'jmEventPrinterState', 'jmEventPrinterStateReasons', and 'jmEventPrinterIsAcceptingJobs' objects.

6) Revised 'Rules for Encoding Notifications' section to add 'jmEventPrinterState' and 'jmEventPrinterStateReasons'.

7) Revised 'IANA Considerations' section to specify there are none - no enumerated or keyword textual conventions are now defined in this document.

8) Revised 'Internationalization Considerations' section to specify that US English keywords are used in 'jmEventTriggerEvent', 'jmEventPrinterState', and 'jmEventPrinterStateReasons' objects and thus no explicit natural language tagging is required.

<draft-ietf-ipp-not-over-snmp-00.txt> - 10 October 1999

1) Initial version.

14. Intellectual Property Notice

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

15. Authors' Addresses

Tom Hastings
Xerox Corporation
701 S Aviation Blvd, MS 834-03E
El Segundo, CA 90245

Phone: +1 310-333-6413
Email: hastings@cp10.es.xerox.com

Ira McDonald
High North Inc
221 Ridge Ave
Grand Marais, MI 49839

Phone: +1 906-494-2434 or +1 906-494-2697
Email: imcdonald@sharplabs.com
Email: imcdonal@sdsp.mc.xerox.com

16. Full Copyright Statement

Copyright (C) The Internet Society (2000). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.