

Minutes of the IPP Working Group meeting in Savannah, GA

(9/25-6 1998)

IPP - Internet Printing Protocol

Tom Hastings volunteer to continue editing the model document following Scott's departure. We need someone to create and edit the implementor's guide. Don Wright volunteered as IPP Web Master. Updates must be sent to Don via e-mail including the subject, and the IPP WEB to be changed.

Review of the Chicago IETF meeting

The Chicago meeting of the IETF occurred the week after our Toronto PWG meeting (end of July). It was determined that for IPP to be accepted as an IETF standard it must reference the IPP URL scheme recommended by our Area Director, Keith Moore. From the IETF perspective, we still haven't published a complete Internet Draft which includes this new URL scheme as mandatory. The concept of a security parameter on the URL, which would allow secure and non-secure printing over the same port, was presented to the Applications Area Directors. While they seemed to like the idea, we are still obliged to fully document and publish a specification of any proposed security method and seek approval of the IETF Security Area Directors. This process is uncertain and unpredictable. It has been one month since the Chicago IETF meeting and there is no resolution regarding the final status of IPP. The TLS (security) group is working on HTTP1.1 over TLS but this work has not been progressing very fast either.

Review of Bakeoff

The IPP interoperability test was hosted by Microsoft the week of September 21 with 16 companies. Of 128 possible unique client/server pair wise transactions, 124 were successful. Additional test suites were exercised. 20 issues were captured. An observation is that there was no "test server" to exercise clients. There may be another bakeoff in mid-Feb 1999 to focus on security, Client contention, Optional operations, and untested attributes. We need to establish detailed criteria prior to a 2nd bakeoff (just wanting to remind ourselves what a fine host Microsoft can be is not an acceptable reason ;-).

A PWG press release will be drafted by Don Wright (Lexmark) for e-mail distribution to as many contacts as we can collect. If you send Don your contacts with e-mail addresses he will reciprocate by sending you the final list of recipients. All will get to see the draft prior to release. The goal is to release by Friday 10/9

Review of the IPP Implementor's Guide

At this stage, the Implementors Guide is basically a list of issues structured into 3 sections

- Model issues

- Encoding and Transport Issues

- Bakeoff (preparations) Issues (removed since bakeoff is now in the past)

It was agreed that the Implementor Guide will take on a larger role with different organization. Much of the explanatory nature of HTTP hints that were found in the Model and Protocol documents will be migrated to the Implementor's Guide along with any FAQ information. We prioritized review of issues resulting from the bakeoff first. The numbers correspond to issues on Carl-Uno's list which is forming the Implementor's Guide/FAQ. Since there was a lot of discussion, I captured what I could.

Issues from the Bakeoff

Minutes of the IPP Working Group meeting in Savannah, GA

(9/25-6 1998)

1.10 - **Case sensitive issues in the URL.** We agree that parts of the URL may be case sensitive and IPP implementations should not change the case. We agreed not to mandate lowercase URLs but this is recommended wherever possible. Server should also be case insensitive if possible. (Note, these agreements were somewhat later revised at the 10/7 conference call to de-emphasize the recommendations for lower case and emphasize the possibility of case sensitivity in path name parts of the URL).

1.11 - **Some implementations do not send an HTTP response to the Cancel-Job operation.** This would be considered a bug and should be fixed in the implementation.

1.12 - **What should happen when there is an attempt to CANCEL a completed job?**

In one place, the model document says you should be able to cancel a job ANYTIME after creation. In another place, the model document says that it's an error if you try to cancel while the job is already in a completed state.

If the job object no longer exists the response to a CANCEL is "no such job". If you issue a cancel and it's too late to do the cancel, you should not say the job was canceled. This is an error.

Right now, only an administrator can set a job to retain. Also, only the administrator can purge jobs from being retained. Canceling a job does not purge a job from the retained state. There is still a lot of confusion about job history, retention states etc.

1.14 - **Recommendation that "queued job count" be implemented by all servers.**

Microsoft's client relies on this attribute and support for it will improve interoperability.

1.15 - **Does "queued-job-count" include pending-held jobs?**

After a lot of debate... Yes. What we really need is a new attribute to also show ACTIVE jobs. This could be added in a future version of IPP.

1.16 - **What if there is an empty job templet attribute group in a Print-Job request?**

If client is doing print job and has no job templet attributes to send does it still need to send the job templates tag?

The client Optionally supplies a set of job templet attributes. Place holder in protocol packet for where the job templet attributes go. Recommend NOT to send a group tag if the group is empty. But servers must accept empty templates and must be able to handle the absence of templates.

1.17 - **May an IPP server omit empty groups if there are no attributes to return?**

If you have no attributes you should not send the tag. Receiver must handle empty or absence. Except Get-Jobs response. See protocol specs. No change.

1.18 - **Unsupported attributes should be returned in the unsupported group.**

We decided a server MAY return the unsupported attributes but is not required to. Thus, a client cannot rely on unsupported attributes from a Get-xxx operation.

1.19 - **When an unsupported char set is requested, what character set should a server use when returning the unknown attribute?**

The server should use it's default character set as currently stated in the spec.

1.20 - **Resolution appears to be 3 Integers with a total of nine bytes.**

The last integer is, for some reason, only one byte. This is an artifact of the protocol document in that it is really an integer, integer and a byte. We should say 3 VALUES to make things clear.

1.21 - **Server is not REQUIRED to fail operations based on order of attributes but may.**

Minutes of the IPP Working Group meeting in Savannah, GA

(9/25-6 1998)

Client should be well behaved. Section 16 of model will be extracted into implementors guide which will potentially be issued as an informational RFC. The contents of the implementors guide should not conflict with the overall suite of proposed IPP standards.

1.22 - If you pause the printer but keep sending data, the printer will ultimately throttle the client.

Some printers might not be able to accept jobs while paused. Most printers will have some limit wherein they will not be able to accept more jobs when paused. This is working as designed. Just fix the test script or emphasize it's context.

1.23 - Returning job-uri and job-id when "job-template" attributes are requested.

Working as designed. Fix the script.

1.24 - Definition of "success-attributes-substituted-or-ignored" and unsupported attribute values.

This error code should be used for any unsupported value, not just for unsupported attributes. We need to clarify that "success-attributes-substituted-or-ignored" pertains not only to JOB operations, but to all operations where requested attributes are not supported. We see that even "successful-ok" is defined to only pertain to job operations. Needs some cleanup.

1.27 - How to staple multiple documents but start each document on a new sheet.

It seems the multi-document behavior is not well specified. What good is a multi document definition where documents are not required to start on a new sheet?

1.28 - What MUST an IPP object do if Create-Job never gets a follow-on operation?

Implementation dependent. Recommend Time-out.

1.29 - What does an IPP printer return in a Print-Job response if the job was canceled by an administrator before the client had supplied all the data and how do you stop the streaming of data?

A (new) status code (job-terminated-during-transmission" or "server-job-canceled") should be sent from the server to the client. Error includes job-id if there is one but there is no guarantee one will have been created. When the client sees the fatal error it should stop transmitting. If the server thinks the client is ill behaved, the server can drop the connection.

2.3 - Can we allow clients and servers to support HTTP1.0 and not HTTP1.1? We needed HTTP1.1 for chunking to facilitate streaming. But, if HTTP1.0 does not require a length on the header, perhaps chunking is not necessary.

You MUST support HTTP1.1 but are not required to reject 1.0 (although you may choose to). Be conservative about what you generate (client - HTTP1.1 preferred)... Be liberal in what you accept (Server HTTP1.0, 1.1). If a client starts off with 1.0, however, the server should not respond with 1.1 chunking.

2.4 - HTTP get on an IPP printer's URI.

Some printers allowed this to access the printer's web page. This is a good idea and should be encouraged in the implementor's guide. The standards should remain silent, however, on this topic. There were two interpretations of this. Port 80 or just exactly the URL of the printer (even if it includes :631). This may require further clarification. This does not override the URI provided in the more-info attribute.

2.5 - Too complex. Handle later.

Minutes of the IPP Working Group meeting in Savannah, GA

(9/25-6 1998)

2.6 - **The possibility of fragmentation of HTTP is a fact of life that clients and servers both must handle.** SUN's client provides a good test vehicle for this condition.

2.7 - **Chunking.**

If you are a 1.1 client you should anticipate the server may chunk responses. There is a way to ask for no chunks. Xerox to document the means. We need to be careful that a server does not use this method to ask the 1.1 client not to chunk. This could break things.

2.15 - **We agree with Carl that IPP specification should not be more specific about protocols we reference (HTTP) than the actual specifications for those protocols themselves.**

We should be aware that RFC2068 (HTTP1.1) is under revision. In the Protocol document, move the section about HTTP headers to the implementors guide. This section has the port 631 in it, however. We should not delete this entirely.

2.16 - **Use of HTTP1.1 as transprot.**

Reference HTTP1.1 RFC.

2.17 - **Support for receiving Chunks is required.**

Some existing web servers do not support this, however

2.18 - **This is another example where IPP should not further restrict the definition of HTTP1.1.**

We accept that "Expect: 100 Continue" is valid although we question whether or not it may have been dropped from the most recent drafts of HTTP1.1. Perhaps the statement "Client must not expect a response form an IPP server until the client has sent the entire request" is entirely pertaining to the IPP layer not HTTP, however... So HTTP1.1 Expect 100 continue is not the answer

2.xx (New) - **Type URL into client w/o port number which port does it mean? 631 or 80?**

Reference Protocol document "Note: even though port 631 is the IPP default, port 80 remains the default for an HTTP URI. Thus a URI for a printer using port 631 MUST contain an explicit port, e.g. "http://forest:631/pinetree".

Other Issues pertaining to IPPv1.0

1.30 - **Small job disappears from the device as the IPP server response is still being constructed.**

The real problem is there needs to be more communication or some latency in job state in the system between the server and device. If the server is confident that the job completed then "Completed" would be a reasonable response. Else, there is no real way to define what the server should say.

1.32 - **Should a Get-Jobs result in a list of all jobs or only jobs submitted via IPP? If both, should the jobs be distinguished?**

The desire (and recommendation) is to list all the jobs and also have all IPP operations apply to all jobs. Thus, a job submitted via LPR could be canceled via IPP, for example. If some IPP operations (like Cancel) require access control, and the user is unknown on the non-IPP job, access would be considered anonymous.

1.35 - **Names for enums?**

Correct the example in the protocol document.

Minutes of the IPP Working Group meeting in Savannah, GA

(9/25-6 1998)

1.36 - **If you get a request ID of 0 (which is invalid) or if the request ID is somehow otherwise unintelligible, then what should the request ID be in the response?**

We need a special value. 0 is not a legal value for request ID so should we return 0? Does a server really have to reject a request ID of 0? This is a MIB issue not HTTP. But what about other forms of corruption? Every IPP request needs a response. The issue is, should you validate request IDs? Randy says you can't have a corrupt request ID. If you get 4 complete bytes, you just return the ID. If you never get to the point where you have received the entire request ID then use 0 in the return.

1.37 - **Value for empty sets.**

"None" in enums is different than "None" in keywords. No change except note in Implementors Guide

1.38 - **Syntax for Boolean.**

Remove the analogy to an enum.

1.39 - **If someone includes "my-jobs" in a request, then it is recommended they supply "requesting-user-name" but this is not necessary.**

It is typical, however, to detect user via the underlying transport.

2.8 - Examples should be fixed. If you know of a problem with an example, send this information to the editors.

2.11 - We agreed to move most of section 4 of the protocol document to the implementor's guide.

2.12 - Leave as is.

2.13 - Connection management recommendations will be moved to the implementors guide.

2.14 - **Relative vs. Absolute URI's.**

We should not require that a server reject the job URI if the request does not (semantically) match the job URI in the transport. In fact, we recommend servers be forgiving under these circumstances even though it is mandatory for the client to provide the job-uri.

1.40 - **Empty attribute and delimiter?**

Some server implementations do not add delimiters for empty attribute groups, and some client implementations assumed delimiters will always be there even if the attribute group is empty. We should clarify. Same issue as 1.17.

1.41 - **Spooling jobs is not a requirement for all IPP printers.** Some tests will not be applicable.

1.43 - see 2.14

1.44 - **Text or name with language is mandatory.** Many IPP implementations did not support text/name with language attributes, and some were crashed when they received "with language" attributes. Implementations should be fixed if they do not support.

1.5 - **Converting unknown characters.** Don't mandate converting unknown characters to question mark. Use whatever the system default behavior is for this condition.

1.28 - **What if Create-Job never gets an Add-Document, Send-Document or 'last-document' is never set to 'true'...**

The IPP object should close the job after some period of time and:

1. For spooling applications - move the job to the 'aborted' state with the 'aborted-by-system' job-state-reasons value set.

Minutes of the IPP Working Group meeting in Savannah, GA

(9/25-6 1998)

2. For non-spooling applications - move the job to the 'pending-held' state with a job-state-reason of "incomplete-job" and an administratively set time-out (between 30 sec and 4 min - see below).
3. As a fallback - move the job to the 'pending' state and print the job? (A form of natural aging)

These instructions should be described in the Implementor's Guide. We are basically addressing system latencies that may occur during the process of performing a Create-Job group of operations. In general, the Create-Job is intended to flow as a rapid sequence of operations without large discontinuities in time between related operations. We should be cautious in defining a tuning attribute, here, and thereby effecting overall system performance. It is that it is not our intent for the server to keep partially constructed jobs on hold for long periods of time. We couldn't actual agree on a figure but we expect it to be somewhere between 30 sec to 4 mins. The real number should be determined empirically and information updated in an implementor's guide.

1.33 - Equality between different syntax's?

When checking for equality or containment (e.g., "IF NOT in the Printer object's 'job-hold-until-supported' attribute ...") is value type considered? Is a value of type 'nameWithoutLanguage' considered equal to a value of type 'nameWithLanguage' if the default language for the context of the 'nameWithoutLanguage' value is the same as the language explicit in the 'nameWithLanguage' value?

Yes, under these circumstances, but not if the defaults are different because then the semantics implied by the values may not match.

Can a 'name' match a 'keyword'?

Yes, possibly, under these circumstances but not in general. (Need clarification on the question).

IF a 'nameWithoutLanguage' value in the appropriate natural language context CAN match a 'nameWithLanguage' value, is there any harm (other than a negligible increase in network bandwidth consumption) in an application promoting ALL 'name' and 'text' attribute values to 'nameWithLanguage' and 'textWithLanguage' values?

No harm... Another way to state the question is if a client sends an attribute then queries it, must the tagging be identical in the response... We said no.

Keywords are intended to be localized by the client. Keywords on the wire are not localized, however. If the server also supports some administratively defined names, the client realizes these are already localized by the server. Administrator has defined a name and the client can supply that either with or without language.

1.34 - **Equality between "natural language" tags.** Is natural language considered when comparing 'name' attributes (e.g., "job-originating-user-name", "media", "job-hold-until-supported")?

No if you are referring to Keywords (which do not have natural language). Yes when comparing attribute values which are Names.

[Assertion: ALL 'text' and 'name' attributes have an associated natural language, either explicitly or implicitly.] If so, how strict is the comparison? Does "en" match "en-us", for example?

Comparison should be strict in general but there is some overlap as a function of the languages themselves.

1.25 - **New groups can be added. We need a mechanism for registering them.**

1.26 - **What about unsupported attribute syntax's?** Does the implementation respond as if the attribute or value were not supported? If so, then Section 3.2.1.2 should add this condition to the list.

Yes - Needs to be explained. Send back the original syntax and the value.

Minutes of the IPP Working Group meeting in Savannah, GA

(9/25-6 1998)

1.31 - Not a problem

2.9 - **Do we need a SIGNED-INTEGER?**

Section 3.2 of the Encoding document provides the following explicit definitions: "SIGNED-BYTE = BYTE" and "SIGNED-SHORT = 2BYTE". For consistency, we might want to have a similar definition for SIGNED-INTEGER (i.e., "SIGNED-INTEGER = 4BYTE").

Fix the document by adding to the formal grammar.

2.10 - **Is it obvious to everyone that the "job-attributes" tag is what needs to be used to signal the start of the job-template attributes in a job submission?** It wasn't to me until I saw an example in section 9.

The protocol document needs a clarification.

Review of new operations

1. When you "Release-Job" you don't know if it went into pending or processing. Should the job state be returned? No. For example, Cancel-Job does not return the job state. If you want to know the state you can query. The rest of the document needs to be edited to remove lines like (99)... Return indicated new job state. (line 59).

2. Printer operations are also specified as returning the new state if state changes. This, should be removed from the spec. (line 285)

3. These changes will be made and the document will be issued for last call in the PWG.

Review IPP scheme

If you start with `ipp://xxx` on the request line, this will be mapped to <http://xxx:631> over the wire. References to URL's in the MIME body, itself, are unchanged. If a client does not follow the IPP scheme and sends HTTP URL's these get returned unaltered in the response.

We are confused about the scenario where, if IPP URI's are generated by administrators and security parameters are part of the URI, then the client must use whatever security is indicated in the printer URL including, possibly, more than one security scheme, if the URI indicates more than one security parameter. For a simpler case, using just TLS, there is no use in trying to invoke the URL following conversion from IPP to HTTP URI, unless TLS is supported. How does the user know the difference, in the directory, between secure and non-secure? It seems like there needs to be two URL's in that case. The lack of clarity on these topics is a clear indication that a security implementor's guide or improved drafts is necessary from the security's group in the IETF.

There was further interesting but rather unfruitful discussion regarding security. While there may be some specific, unique printing related security requirement which we should be addressing, general HTTP security is beyond the scope of the IPP working group as a whole.

We are waiting for a draft for TLS over HTTP which hasn't surfaced. We need to review the latest DAA and HTTP drafts. These specs tend to be rather long and complicated and difficult to understand. It will obviously take some time for us to digest, prototype and then bakeoff these items.

Minutes of the IPP Working Group meeting in Savannah, GA

(9/25-6 1998)

Review of IPP/IFAX (Evening)

Ron Bergman
Tom Hastings
Carl-Uno Manros
Bob Herriott
Harry Lewis
Lee Farrell.

In Chicago, at the IETF IPP meeting, there was some talk about running IFAX over IPP to avoid problems with real time fax over e-mail. We need to develop an informational paper between the two working groups. We need to understand the problem and help with the recommendations. Paper provided by Richard Shockey
Draft-ietf-ipp2ifax-map-01.txt. Can we really notify that the job was printed? They can query.

FAX negotiates prior to sending. They don't get this with e-mail or get it very slowly.
FAX gives confirmation regarding print status. Again, e-mail is slow and unreliable for this purpose.

IFAX has defined and registered MIME types for a couple new TIFF formats (B/W and Color). We can use those as is.

With a Get-Printer-Attributes, an IFAX client could discover which TIFF formats are supported.

There are other things fax has that may seem foreign to IPP, like distributions lists. Some of these may be viewed simply as client issues.

Separate mailing list. Who will host? PWG or IFAX.

Ron Bergman will send our notification paper to the IFAX group.

If the device advertises itself as an IFAX capable, it must support the required TIFF formats to assure some base level of negotiation .

If FCC requires date, name and phone number on each page, we have to determine whether the sender puts this on or the receiver and how to handle potential margin or page scaling issues.

IPP does not mandate creation or completion times. This would be considered a serious deficiency for IFAX. Any IPP printer that supports IFAX would be required to have a real time clock to provide this information. On the other hand... The generator of the FAX is the one that probably needs the clock.

In some areas we will need to align terminology - such as "redirection".

If IPP printer is acting as an off ramp to FAX distribution via GSTN... Then the list of names or numbers could just be tagged in the URL.

FAX might also carry additional compression requirements.

Do we assume all legal requirements for FAX pertain to IFAX over IPP even if the GSTN is not involved in any way?