WEIRDS WG
IETF 84 – Vancouver, CA

Olaf Kolkman <olaf@nlnetlabs.nl>
Murray Kucherawy
<superuser@gmail.com>
NOTE WELL

The brief summary:

• This summary is only meant to point you in the right direction, and doesn't have all the nuances; see below for the details.

• By participating with the IETF, you agree to follow IETF processes.
• If you are aware that a contribution of yours (something you write, say, or discuss in any IETF context) is covered by patents or patent applications, you need to disclose that fact.
• You understand that meetings might be recorded and broadcast.

The details:

For further information, talk to a chair, ask an Area Director, or review:

• BCP 78 <http://tools.ietf.org/html/bcp78> (on the IETF Trust), and
Agenda

• Welcome and Administrivia [0:05]
• Part A: Some Guiding Principles
  – Document Organization [0:10]
  – Supporting Objects [0:05]
  – Service Differentiation [0:30]
  – RESTful Versioning and Extensions [0:30]
  – URI Templates [0:10]
  – Object Inventory [0:30]
  – Resource Discovery [0:30]
• Part B: Specifics (if any) [0:??]
• Part C: Next Steps [0:15]
• Any other business [0:??]
Administrivia

• Welcome!
• Blue sheets
• Jabber scribe
• Minute-taker
Document Organization

• We would like to build a set of documents from the bottom up, and resist divergence until as close as possible to the top

• Low level:
  – Requirements (maybe)
  – Common transport and authentication
  – Common query “language”, URI construction
  – Objects common to all clients and servers

• High level:
  – Names objects vs. Numbers objects
Supporting Objects: “MUST” support

• When we say a server MUST support something, we mean it MUST NOT reject a request that meets our standard syntax, buy MAY ignore it.

• When we say a client MUST support something, we mean it MUST NOT consider the presence of a standard field to be an error, but MAY discard its content.
Service Differentiation

• How do we accomplish this?
  – Personally Identifiable Information should only be shown to authorized parties
  – Rate limiting

• [Shuo Shen presentation]
RESTful Versioning and Extensions

• Do we need to version the protocol?
  – In terms of supported objects

• How do we go about ensuring extensibility?
  – IANA?

• [Andy Newton presentation]
URI Templates

• RFC6570
  – Client contacts the server to ask for a template that describes how to form a query URI
  – Client fills out the template based on an internal table of objects and their values
  – This produces the URI, which is then queried

• Client complexity!

• But this is “the way”, according to web experts
  – Standards don’t own the URI space, servers do

• Also means that a client doesn’t need to be specifically designed as a names client or numbers client
Object Inventory

• We need to define the set of data that will go into our first standard set of objects
  – Common vs. names vs. numbers
• [Ning Kong presentation]
• Design team approach
Discovery

- So you have a question to ask. Where do you start?
- Is mapping in scope? If so, should we limit the scope?
  - What ideas are on the table here?
- [Chris Wright and John Levine presentation]
Specifics

• Do we have any other specific topics to discuss before the design teams get cracking?
Next Steps

• Which documents do we adopt, or create?
  – Who will act as authors/editors?
    • Individuals
    • Design team(s)
  – What milestones shall we set?