Authorization and Authentication issues

As I see it, there are three main areas where we need to think about issues of authentication and authorization in Sakaibrary.

1.1 Citation permissions/scope. Who can view/edit/embed a citation or citation list?
1.2. Auto-import by URL. How do we make sure that only a valid user can push information into a citation (as we're planning with Google Scholar).
1.3. Licensing issues. How do we make sure people only access what they're allowed to access?

I'll take them in turn, below.

Citation permissions and scope

These are essentially policy issues, but of course they have significant effects on the utility and usability of the tool. I’ll explore what I see as the major options below, with comments on the pluses/minuses from a user perspective.

Option 1: Citations belong to a user across courses

In this scenario, a citation belongs to a user. There's an "uber-list of all her citations in all courses, and a single "Citation List" (as we currently use the term) would be a subset of the uber-list. Each user would have a single instance of a given citation object that might be used in more than one course within the Sakai instance.

The utility of this implementation assumes we provide the ability to list/search ones own uber-list and copy (references to) a citation object from one list into another.

Benefits: users tend to take many courses within a discipline (their major) and would be able to maintain a full record of their citation history. Users won't have to repeat searches or import records simply to get a citation they've already used in another course. Anything that uses a citation would still add to/draw from the uber-list, maintaining a consistency of location for where these things ultimately live.

Drawbacks: Requires a backend that exists essentially independent of the course structure. May mean that an object (a citation list) could be owned by Person A while its constituent parts (the individual citations) are owned by a series of other people. Could get messy quickly with the permissions system already in Sakai.

Questions: Don't we want people to use RefWorks or some such for this sort of full-on bibliographic maintenance? Is there a mechanism for saving this data from one semester to the next?

Option 2: Citations belong to a user within a specific course context

As above, but restricted to a single course. Less useful in that users might have to repeat searches/imports for each class to get common citations.

Option 3: Citations belong to the course

Here, there is no personal ownership of a citation (although it would be great if there was a way to have personal authorship for metadata associated with a citation, e.g., each user could have their own notes, tags, keywords, etc. without changing the core metadata like volume, title, etc.).

In this situation, you might have a collection (a citation list) owned by an individual while the citations themselves are owned collectively by the course.

Benefits: supports collaborative citation list building. Keeps focus on the course.

Drawbacks: means people can mess with citations that everyone is depending on (with no audit trail??). If we're focussed on supporting this type of pedagogy, should probably provide mechanisms to determine if a citation has already been added by a classmate to avoid duplication. Will need ways to make sure editors don't overwrite existing work.

Option 4: Citations belong to a specific citation list object

This is the gestalt of what we have now. Citations can't be moved (or even copied, at the moment) from one list to another context. Ownership is based on ownership of the list, and only the person who owns the list could edit a citation in it (this is not what we have right now, but probably should be).

Benefits: Probably the easiest to implement.

Drawbacks: Much more limited pedagogical use than the previous; makes both reuse and collaboration difficult.

Auto-import by URL

I'm using "auto-import by URL" to describe the process by which a carefully constructed URL of the form:
http://<machine>?<target object>&<citation information>

...is used to “push” citation metadata into an existing target within a particular Sakai instance, as we're planning on doing with Google Scholar. This is in some sense a solved problem in that RefWorks has similar functionality. It's not, however, solved by us.

The <target object> above is a unique identifier describing the particular Sakai object to which the citation information will be added. Depending on what we decide about citation scope, it could be a particular citation list, an uber-list, or a combination of a user’s (or course's) uber-list and a secondary object to which a reference to the citation should be added (e.g., "add to dueberb's uber-list and also append a reference to Citation List 11212").

The authorization issue here is that we need to make sure that only an authorized person can add stuff via this mechanism.

An initial, easy-ish solution is that we use Sakai’s auth and require that the user start in Sakai. Example:

- User logs into Sakai. A cookie is set with her auth information
- User goes to a specific citation-container.
- User clicks on a link and goes to Google scholar, sending along <target object> information specific to the citation container, but no auth information.
- Google scholar does the search and creates a link like that shown above, with <target object> and <citation information> embedded in the cite.
- User clicks on that link, which leads her back to Sakai. Sakai then determines, based on the existing Sakai auth cookie and the <target object> given, whether or not this user can add the citation in this context.

We've essentially given all auth responsibility to the Sakai installation and are relying on the user to always start from within Sakai so the <machine> and <target object> data can be sent to Google.

The other, more complex option, is to create a "generic" link to a Sakai instance and then ask the user for auth and target information. So:

- User searches Proquest and hits the UMich SFX link
- SFX provides a generic link to CTools for this reference
- User is challenged for login information (if necessary)
- User is presented a list of possible targets to which she has write permission
- User chooses one and the citation is added.

This is obviously more complex, not least because it requires all citation target objects to register themselves with the citation service. If we're only talking about citation lists, that's one thing, but if everybody and their brother can add citations (e.g., Syllabus), then things get harder.

**Licensed Library Content**

This is the auth issue about which we've talked the most. A useful simplification of the way things work is:

- Much (most?) of our licensed library content is restricted based on the IP of the calling machine.
- On campus machines "just get access," with the (usually erroneous) assumption that only authorized users can get access to on-campus machines.
- Any request by an off campus machine is routed through a proxy service, which challenges the user, checks her permissions, and then (with its on-campus IP address) takes care of bridging between the user and the database vendor. The proxy server is trusted (by the campus and the vendor alike) to restrict access to only those who deserve it.

The problem is that the Sakai server has an on-campus IP, and so is never challenged, meaning that anyone who has access to Sakai has access to the licensed content.

The challenges are:

- There isn’t a one-to-one mapping between Sakai groups and library access, nor can we usefully force this issue.
- Single-signon systems tend to use cookies for session information, and the Sakaibrary service won't be dealing with either cookies or the necessary names/passwords for individual users.
- We likely can’t assume a common format for querying directory services to see what a user can do, or even the existence of such a general service.
- I'm pretty sure we can't assume that a metasearch engine has a "superuser" who can mask as other users (in a "log in as root, but do things as if I were dueberb" sort of way). Even if they do, we still need to get lists of targets to which a specific user has access.

The question is, given only the information I assume we can get from Sakai (the Sakai username and the IP of the user), can we figure out what a user is allowed to do?

Ideally, it'd work like this:

- User logs into Sakai and starts Sakaibrary
- Sakaibrary takes username and IP address and queries some service to find out what this user has access to.
- User is presented with a list of possible search targets tailored to her own access.
- User does search, and we either again check the sources or maybe just the referrer to make sure she didn't hand-construct a query to sources she shouldn't be accessing.
The service to be queried is going to differ radically from campus to campus as ways of identifying sources and ways of authenticating/authorizing users vary; hence we should probably define this as a web service.

**Web services for user search permissions / targets**

What I'm envisioning is a web service that takes search information and returns a set of searchable targets in a common format. The search information may include things like keyword or category, but would certainly have to include username and IP.

The return value would be a list of keys that uniquely identify search targets. Sakaiibrary could then query for the full information associated with each search target, or (more likely) consult a local cache of that information to build up categories, choice lists, etc.

A default super-user (root, the empty username, whatever) could be used to indicate that permissions should be denied. In this mode, we'll simply get back a list of search targets that can be accessed by *some* user, useful for building up the aforementioned local cache.