What's cooking in MediaWiki

Single sign-on, version tagging, and other goodies

Brion Vibber
Wikimedia Foundation, Inc.

August 5, 2006

Wikimania
Cambridge, MA
Too many accounts!

- Each language and project has its own set of user accounts
• This makes cross-wiki operations like uploading an image to Commons harder:

  • log in again

  • have to create your account a second (third, fourth) time

• Sometimes a username belongs to different people on two project or language wikis, creating confusion
Immediate goals

- Let the same username and password work for logging in on any Wikimedia project wiki, once created a single time
- Let the same username belong to the same person on all Wikimedia project wikis
Long-term goals

• Only have to log in once per session for all Wikimedia wikis

• May require some fancy cookie work, so likely not immediately active

• Use unified accounts to support inter-site protocols such as OpenID authentication cleanly
Why doesn’t it work like this to start with?

- Sad answer: nobody got around to central user database support until hundreds of separate wikis were already in place.

- Transition pain:
  - converting database entries
  - merging accounts
**Transition!**

- Basic user data collected into a transitional database to be merged...

<table>
<thead>
<tr>
<th>lu_dbname</th>
<th>lu_id</th>
<th>lu_name</th>
<th>lu_email</th>
<th>lu_editcount</th>
</tr>
</thead>
<tbody>
<tr>
<td>enwiki</td>
<td>1</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>4200</td>
</tr>
<tr>
<td>frwiki</td>
<td>30</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>1337</td>
</tr>
<tr>
<td>jawiki</td>
<td>500</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>87</td>
</tr>
<tr>
<td>srwiki</td>
<td>700</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>0</td>
</tr>
</tbody>
</table>
Assign an account owner

- The global account is assigned first to the owner of one of the local wikis’ accounts.
- In most cases, this will just be the account with the most edits.

<table>
<thead>
<tr>
<th>lu_dbname</th>
<th>lu_id</th>
<th>lu_name</th>
<th>lu_email</th>
<th>lu_editcount</th>
</tr>
</thead>
<tbody>
<tr>
<td>enwiki</td>
<td>1</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>4200</td>
</tr>
<tr>
<td>frwiki</td>
<td>30</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>1337</td>
</tr>
<tr>
<td>jawiki</td>
<td>500</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>87</td>
</tr>
<tr>
<td>srwiki</td>
<td>700</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>0</td>
</tr>
</tbody>
</table>
E-mail match

- Any accounts with matching e-mail are next attached automatically.
- Passwords can be reset by e-mail, so any account with matching e-mail can be accessed by the same person.

<table>
<thead>
<tr>
<th>lu_dbname</th>
<th>lu_id</th>
<th>lu_name</th>
<th>lu_email</th>
<th>lu_editcount</th>
</tr>
</thead>
<tbody>
<tr>
<td>enwiki</td>
<td>1</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>4200</td>
</tr>
<tr>
<td>frwiki</td>
<td>30</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>1337</td>
</tr>
<tr>
<td>jawiki</td>
<td>500</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>87</td>
</tr>
<tr>
<td>srwiki</td>
<td>700</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>0</td>
</tr>
</tbody>
</table>
Merge unused accounts

- Some accounts have never actually been used for editing.
- These can be reclaimed with little conflict.

<table>
<thead>
<tr>
<th>lu_dbname</th>
<th>lu_id</th>
<th>lu_name</th>
<th>lu_email</th>
<th>lu_editcount</th>
</tr>
</thead>
<tbody>
<tr>
<td>enwiki</td>
<td>1</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>4200</td>
</tr>
<tr>
<td>frwiki</td>
<td>30</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>1337</td>
</tr>
<tr>
<td>jawiki</td>
<td>500</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>87</td>
</tr>
<tr>
<td>srwiki</td>
<td>700</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>0</td>
</tr>
</tbody>
</table>
First login: password check

- The next time the person logs in, we can merge any remaining accounts with matching passwords.
- Encrypted password storage means we can’t compare them until the user types it in for us.

<table>
<thead>
<tr>
<th>lu_dbname</th>
<th>lu_id</th>
<th>lu_name</th>
<th>lu_email</th>
<th>lu_editcount</th>
</tr>
</thead>
<tbody>
<tr>
<td>enwiki</td>
<td>1</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>4200</td>
</tr>
<tr>
<td>frwiki</td>
<td>30</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>1337</td>
</tr>
<tr>
<td>jawiki</td>
<td>500</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>87</td>
</tr>
<tr>
<td>srwiki</td>
<td>700</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>0</td>
</tr>
</tbody>
</table>
Yay, done!

- If everything could be matched like this, the accounts have been merged totally transparently.
- Just keep logging in like before, but now it works on wikis you haven’t visited before, too.

<table>
<thead>
<tr>
<th>lu_dbname</th>
<th>lu_id</th>
<th>lu_name</th>
<th>lu_email</th>
<th>lu_editcount</th>
</tr>
</thead>
<tbody>
<tr>
<td>enwiki</td>
<td>1</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>4200</td>
</tr>
<tr>
<td>frwiki</td>
<td>30</td>
<td>Brion</td>
<td><a href="mailto:brion@wikimedia.org">brion@wikimedia.org</a></td>
<td>1337</td>
</tr>
<tr>
<td>jawiki</td>
<td>500</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>87</td>
</tr>
<tr>
<td>srwiki</td>
<td>700</td>
<td>Brion</td>
<td><a href="mailto:brion@pobox.com">brion@pobox.com</a></td>
<td>0</td>
</tr>
</tbody>
</table>
Conflicts resolution

• It won’t always go smooth, of course... We have to resolve conflicts if there are accounts with:
  • missing or different email address
  • AND different password
  • AND have been used for edits
Where do conflicts come from?

• Registered by same person, but didn’t keep passwords and email synchronized

• OR registered maliciously by an impersonator

• OR registered by a different person legitimately
Resolving false conflicts

- Accounts that are owned by the same person but couldn’t be matched automatically can be resolved on login:
  - Confirm you know both passwords by logging in
  - OR reset the password by e-mail
Resolving true conflicts

• This is harder, but possible:
  • On login, the minority account holder will have the option of renaming their account so it no longer conflicts
  • After some time has passed (weeks?), remaining non-matching accounts may be forcefully renamed.
Single user status

• This has been delayed a long time because every time devs discuss this, it becomes a big fight. Details are non-negotiable or we’ll get sidetracked again!

• Basic back-end login and auto-transition code working

• Should be ready for live transition tests later this month, and can go live afterwards

• May delay final transition until Wikimedia board election is complete to avoid breaking votes
Stable revisions

tagging isn’t always a crime
Visibility control

• **We** know that Wikipedia is a work-in-progress and vandalism is temporary and harmless

• but **random visitors** to the site don’t know this!

• A short time-delay or sometimes explicit approval before updating the view could perhaps reduce the impact of vandalism

• Keep it easy to find and fix with a visible notice of new, hidden versions
Harvard Law School

Harvard Law School (HLS) is one of the professional graduate schools of Harvard University. It is one of the world's leading law schools, with about 1,800 students in Cambridge, Massachusetts. It is home to the largest law library in the world.

HLS routinely places as one of the top three schools in rankings, along with Yale Law School and Stanford Law School, and usually receives the highest reputational scores from judges, academics and practitioners. According to U.S. News & World Report, its entering class boasts the highest LSAT scores in the nation and its graduates have a higher bar passage rate and average salary upon graduation than do graduates of its closest
Revision tagging

- Planned for experiments on German Wikipedia
- Possible uses:
  - Marking stable versions for display
  - Marking reviewed versions for publishing subsets
  - Allowing edits on locked articles while default visitor view is to the last-checked version
Neat? What’s so hard?

• Templates and images!
  • May be vandalized or changed incompatibly since the original version was marked
  • A stable revision with a bunch of vandalism in a template... not so great.
How can we solve this?

- Track the revision ID of the tagged page version...
- **and** the current revision IDs of each template used
- **and** the version of each image used
Err, what’s that mean?

- Unlike protected pages today, tagged versions will always display the same...
- even if someone changes the templates and images it uses.
But who makes the decisions?

• Aye, that’s a good question.
• Most likely, ‘protection’-style tagging will be done by sysops like today
• Publishing tagging may be done by ad-hoc groups?
• It’s still in experimental planning, details up in the air...
What about ratings?

- Ratings or reputation systems could become part of how taggable revisions are identified, though automation might be too scary.
Revision tagging status

• Some back-end proof-of-concept code has been done to confirm the template versioning can work

• UI mockups for versioning badges

• Still needs final decisions on how it will be used

• Depending on developer load and community excitement, will see the light of day in one to three months
OpenID

multi-site authentication
• use your Wikimedia account to identify your posts on blogs and wikis supporting OpenID

• use your blog or wiki account to identify your edits on Wikimedia wikis

• a step up from anonymous IP addresses
OpenID status

• Evan Prodromou’s OpenID extension code available in Subversion
• Live on Wikitravel
• Planned to go live on Wikimedia sites after single user unification is complete