Consent Implementation Checklist

Keith Wessel, University of Illinois at Urbana-Champaign

Phase I: Get started

1. (policy) Assemble your team: should include your data owners, privacy staff, UI design staff, and IdP operators. Keep it as small as possible.
2. (technical) Demo the consent feature: show them what you're doing by showing them what you're doing. Demo a login to a service with the default consent page enabled. Apply minimal branding if it makes people happy, but don't put any other work into the appearance.
3. (policy) Explain what you need: once they know how it works, let each team member know why they're on the team and what specifically is needed from them.

Phase II: consent specifics

4. (policy) Select consent rules: break services into categories such as federation, entity tags, etc. For each, decide what will be released and which populations will be prompted to consent if any. This should be done individually by the IdP operator.
5. (policy) Create a table: put the above decisions into an easy-to-digest table.
6. (policy) Present to the rest of the team: you can do this individually or all together. Review each decision for each category and update accordingly.
7. (policy) Identify exceptions: list specific services that are exceptions to the category they're in. Determine how they should be handled. This might produce additional categories.

Phase III: Interface design

8. (technical) Blacklist geeky attributes: identify attributes that are meaningless to end-users and not necessary to consent such as eduPersonTargetedID. Blacklist them from display.
9. (technical) Give attributes friendly names: for the remaining attributes known to the IDP, give each a name meaningful to end-users. Get sign-off from UI design staff for your name choices.
10. (policy) Decide consent options: work with the team to decide if, in addition to do not remember and remember for selected SP, the always consent and don't ask again option will be available. If so, pick wording for this option that makes everyone happy. Also, decide which option will be selected by default.
11. (policy) Decide if per-attribute consent is allowed: work with the team to decide if users can select which attributes to release or if it's going to be an all-or-nothing thing. Note that choosing to not release required attributes can lead to a negative user experience.
12. (technical): Brand and design the consent screen: Create the text and appearance of the consent screen. This is generally a good task to hand to the UI designer then get sign-off from the rest of the team.
13. (technical) Determine what users will see if they choose not to consent: if there's a button on the consent screen to decline consent, create the text and page users will see if they select it. Another good task for UI design.

Phase IV: Consent storage

14. (policy) Determine if consent should be per-device: per-device consent means users will see the consent page more often, but it also allows for cookie-based storage which means no need to add an IdP back-end storage dependency.
15. (policy) Determine number of consent decisions to store and lifetime of decisions: for non-per-device storage, number of decisions to store isn't necessary. For both cases, work with the team to choose the frequency that users should be prompted to re-consent for remembered decisions.
16. (technical) Configure the IdP with the decisions from above. If using non-per-device storage, choose a data store that's highly available.

Phase V: Configure triggers for consent flow

17. (technical) Choose how rules and exceptions will be created: The IdP allows for several ways to create conditions to trigger the consent flow including Spring beans and Javascript. Choose the best method that matches your rules and their complexity.
18. (technical) Determine how exceptions will be configured: exceptions to category-based rules can be hard coded into the config, but this won't work well for on-the-fly changes as the IdP must be restarted to make changes. If dynamic changes are needed, consider reading exceptions in from a file or use entity tagging of exceptions in the metadata provider config.
19. (technical) Create the rules: based on previous decisions, code the conditions for when the user should be presented with the consent page. Of course, test all thoroughly.

Phase V: Document and announce

20. (technical) Document for end-users: Create a useful knowledge base article or web page explaining what user consent and information sharing means to users and the consequences of declining consent. Link it from the consent page so users can get more information.
21. (technical) Publicize: communicate to support staff and, if appropriate, end-users well in advance about the new feature of the IdP. Consider a demo.

Phase VI: launch and tweak

22. (technical) turn it on: enable your new consent flow.
23. (technical) Find what you missed: chances are you've missed some exceptions. Watch your logs to see what services are being accessed that are prompting consent and shouldn't be or the other way around.
24. (technical) Communicate with your help desk: see what, if any, issues your support staff are hearing from end-users. Work with your team to address them.
25. (technical) Report and measure: it might be useful to use logs or, if you're using a back-end database, database queries to see how many users are choosing to have consent decisions remembered and for how many services. If you enabled the don't ask again feature, measure how many users are choosing that. Make these numbers known to your team and to management, it's a good measure of success.