OVP: Website, Submissions, and Processes

Lincoln Lavoie

Background

Current OVP site: https://nfvi-verified.lfnetworking.org/#/

- Code: https://gerrit.opnfv.org/gerrit/admin/repos/dovetail-webportal

Custom web-portal, allows logged in users to upload results, provides basic structure for review, results storage, and publishing of certification records.

- But, all that code must be maintained. For example, supporting new results formats requires updates. Migration to Python 3 will require updates (still requires 2.7), etc.

Desires for OVP

› OVP submissions are form or template driven
  › Some level of completeness or smoke test will help submitters and reviewers
› Test results should only visible to submitter and designated reviewers
› Public Kanban or tracking board displays the state of all submissions
› OVP issued badges are displayed as another tab on the generic LFN landscape view
› Retain a single website for directions and badging process/value proposition (with embedded landscape view for badges only)
  › This website should also link to the appropriate read the docs for each program test plan, additional documentation, etc.

OVP Process (10k ft view)

1. Someone runs some testing, decides to apply for a badge.
2. Test results are submitted, along with company & product information for review.
3. Reviewers ensure test results are complete, follow up with questions, if necessary.
4. Approved results are saved, company & product information used to create public listing on website for badge.

Does this workflow sound familiar?

› Code → PR → Review → Merge

Can We Use Existing Tools?

› Results Submission, Check, Review, and Storage
  › Git Repo
  › Workflow could utilize Issues to track submissions (note, issues can include templates, i.e. the form for company/product info)
  › Tester creates the PR with their results and assigns reviewers
  › Repo's CI process triggers a smoke test, which could check the results for formatting, completeness, etc.
› OVP Listings
  › Landscapes could provide an easy, graphic reference for awarded badges
    › https://github.com/cncf/landscapeapp
  › Easily maintained, via yaml files
  › But, they have limited direct content, and would need to point back to the a repo file or mark down for product info (SUT)

Questions that need answers

› Where is the repo hosted, gitlab, gerrit, etc?

› If results are stored in a repo, who has access to that repo?

  › If limited access to the repo, how are submissions made?

› Can the Landscapes App including a link to the product info, or only the company info from Crunchbase?

› If not Landscape, is there something else or similar?

› Could the badge listing be built directly in a repo as markdown?

Landscape References

› The App: https://github.com/cncf/landscapeapp
› LF Landscape :
https://lf-landscape.netlify.app/category=lf-networking&grouping=category&zoom=150
  › Repo for LF Landscape: https://github.com/jmertic/lf-landscape
› CNCF Landscape: https://landscape.cncf.io/
› https://landscapes.dev/