Survey of University Server Software

Chris Sobczak

April 4, 2021

Abstract

The goal of the survey is to determine the market share of that linux and other open source operating systems occupy in the academic setting. So the figure of interest is the proportion of machines that universities run for thier websites, mail servers and other digital services that run an open source operating system.

Background

Universities spend millions of dollars each year on licenses for their students' software offerings and also for their own infrastructure. Increasing costs for post-secondary education are making it less excessible and a very obvious way to reduce costs is find free and open source alternatives to those expensive licenses. **(cite some study showing costs and how much licenses contribute to the overall cost of attendance).**

In addition to the problamatic cost of closed source, proprietary software, it is also less secure than the open source alternatives **(cite study about this or reference the OpenBSD claims etc).** Especially at public institutions, all of the software that staff and students are required to interact with should also be open source so for the sake of privacy and security, universities should not be given the exclusive access to a specific class of student.

So in this study, I would like to establish a base-line understanding of the proportion of free and open source servers being used by universities around the world. **(look at poststratification)** Are there specific countries whose universities use more open source servers than others? What kinds of services are associated with certain operating systems?

Methods

In general, the sampling unit will be the name of a school. Each school owns one or more domains, and from that can be extracted all or most of the associated subdomains for all the different services that the school might offer from their website, for example email and registrar services. Using cluster sampling, an SRS is taken of all the schools in the sampling frame (Hipo 2021) and then all subdomains are extracted from the sampled schools' registered domains. Each school selected is a **primary sampling unit** (psu) and each of the thousands of extracted subdomains are **secondary sampling units** (ssu). A census of the ssus will be taken, collecting a list of the available services (ssh, smtp, ftp/sftp, ...), the operating system (Microsoft, Debian, ...) and the server software (nginx, Apache, ...) running at each address. **Inevitably, there will be some uncollectable addresses and the effects of this will be mitigated with ....**

My sampling frame is from a github repository containing a json file of all known domains registered by universities around the world (Hipo 2021).

Survey Design

Cluster sampling ... how to mitigate non-responses.

For example, Simon Fraser University (SFU) has `sfu.ca` registered with the Canadian Internet Registration Authority (CIRA) and they have thousands of subdomains under `sfu.ca` such as `mail.sfu.ca`, `canvas.sfu.ca` and `go.sfu.ca`. A lot of the subdomains do not contain relevant web content or are otherwise unused, so to filter out these subdomains before trying to identify the operating systems being used to run the surver at that address, I have used the `host` DNS lookup tool and other methods for checking if there are actual services running at that address. More about the tools I used in section 5 Tools.

Sampling Frame

Sample Size Selection

$e = 0.03$ and $\alpha = 0.05$ Lohr 2019

We want to estimate the proportion of university servers run an open source operating system using a 95% confidence interval with a margin of error of 0.03.

Lohr 2019 page 47: "In surveys in which one of the main responses of interest is a proportion, it is often easiest to use that response in setting the sample size. For large populations, $S^2 \approx p(1-p)$, which attains its maximal value when $p = 1/2$. So using $n_0 = 1.96^2/(4e^2)$ will result in a 95% CI with width at most $2e$."

$$n_0 = \frac{z^2_{\alpha/2} S^2}{e^2} = \frac{1.96^2(\frac{1}{2})(1-\frac{1}{2})}{e^2} \approx 1067$$

No need to use the fpc adjustment since the sample size is reasonable compared to the population size.

Sample Selection

Some of the schools in my sampling frame had more than one associated domain, so when I expanded all the sampled domains into one file, one domain per line, I ended up with 1087 domains to attempt to extract subdomains from.

When extracting the subdomains, my script only processed 1079 domains. Three of these domains that were not processed were `aloma.edu`, `student.uts.edu.au`, and `www.clcmn.edu`. In the case of `aloma.edu`, this is just a typo in my sampling frame for the Alamo Colleges domain, which naturally is corrected to `alamo.edu`. The next missing domain `student.uts.edu.au`, for the University of Technology Sydney in Australia which is just a subdomain for their website `uts.edu.au`. Finally, `www.clcmn.edu` for Central Lakes College-Brainerd is another subdomain for their college that was already extracted from `clcmn.edu`.

In the list of domains, there were 16 duplicated domains and this also accounts for the difference in number of domains that went through the `gen-domains` script and the original sampled list. Here is a list of the duplicated domains: `aku.edu`, `bashedu.ru`, `most.gov.mm`, and `uwo.ca`.

After removing the duplicated domains and generating the domains for the corrected `alamo.edu`, I ended up with a full set of 1081 lists of subdomains.

In the github repository, the file `undup-domains` is the audited file containing all of the highest level domains for the sample.

The next step will be ensuring that two related domains did not go through the process, for example making sure that we did not run `gen-domains` on `sfu.ca` and `mail.sfu.ca`, since this would likely result in duplicated information.

Using the line `cat subdomain/* | uniq -d` we can see that there were no repeated lines.

Results

The full results dataset of proportion of servers at each psu is available at the **github names ...**.

Discussion and Conclusion

Any anticpiated shortcomings in your study desgin and their impact on your conclusions about the questions of interest.

- Non-responses
- Shortcomings in the sampling frame (may not actually contain 'all' schools in the world)

Tools

For taking the raw json file and selecting the sample, I used R and the `rjson` package. All project source files can be found at this GitHub repository. The R script outputs the base second and third level subdomains into a file with one domain per line, for which I run `findomain -t`. This tool takes a domain and searches various databases and tests the domain for subdomains associated with it. I take this and output all the subdomains into a file for each school, and then test if the service is up.

Documentation for the tools can be found in the Appendix

Extracting Info From SSUs

```
curl -I
```

References

Hipo (2021). *university-domains-list*. `https://github.com/Hipo/university-domains-list`.

Lohr, Sharon L (2019). *Sampling: Design and Analysis*. eng. 2nd ed. Chapman & Hall/CRC Texts in Statistical Science Series. Milton: CRC Press. ISBN: 0367273462.

Appendix