Open source risks: perception & mitigation

Petr Špaček, Victoria Risk

2024-05-23

pspacek@isc.org, vicky@isc.org

Experimental session

- Survey
- BIND 9 DNS server
  - as an example project
- Discussion

Survey

https://ec.europa.eu/eusurvey/publication/RIPE88OpenSourceWGSurvey

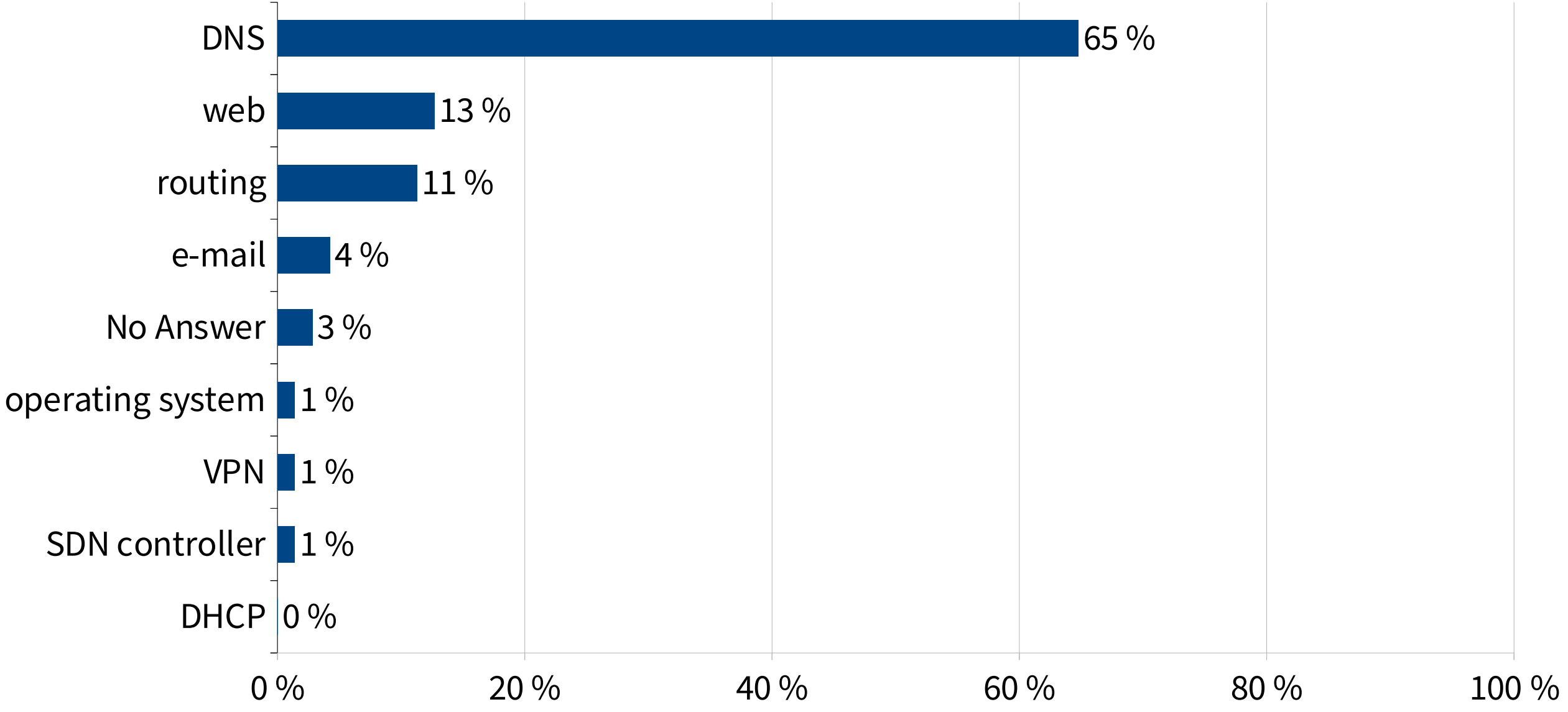
Survey

- What makes a project trustworthy?
  - "Software you consider **mission critical** in your deployment"
- Secure deployment practices
- Risk mitigation practices

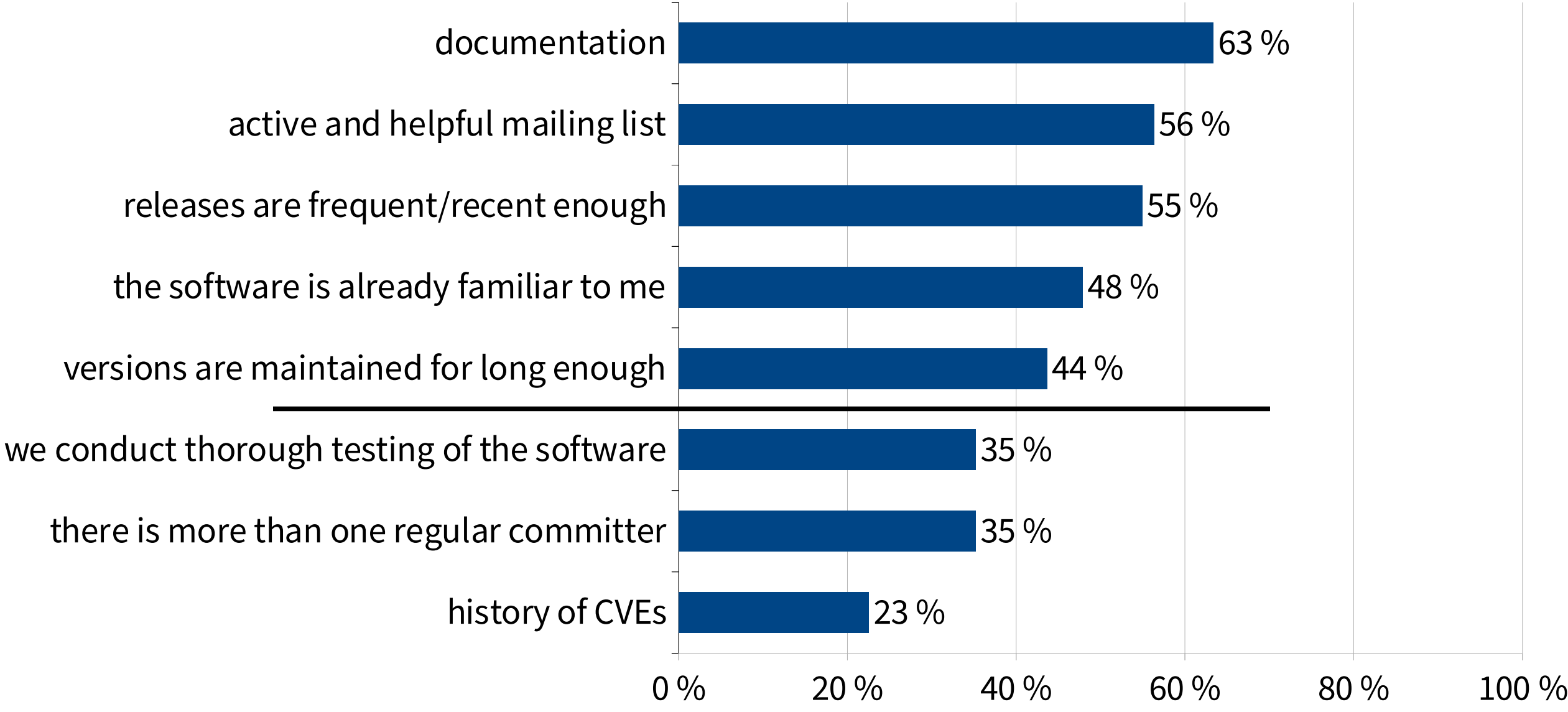
Survey

- Audience – bias
- Operators "who care"
- Presumably experts
  - RIPE Open source WG
  - RIPE DNS WG
  - dns-operations list @ DNS-OARC
  - Internet Systems Consortium's public channels
- 71 answers

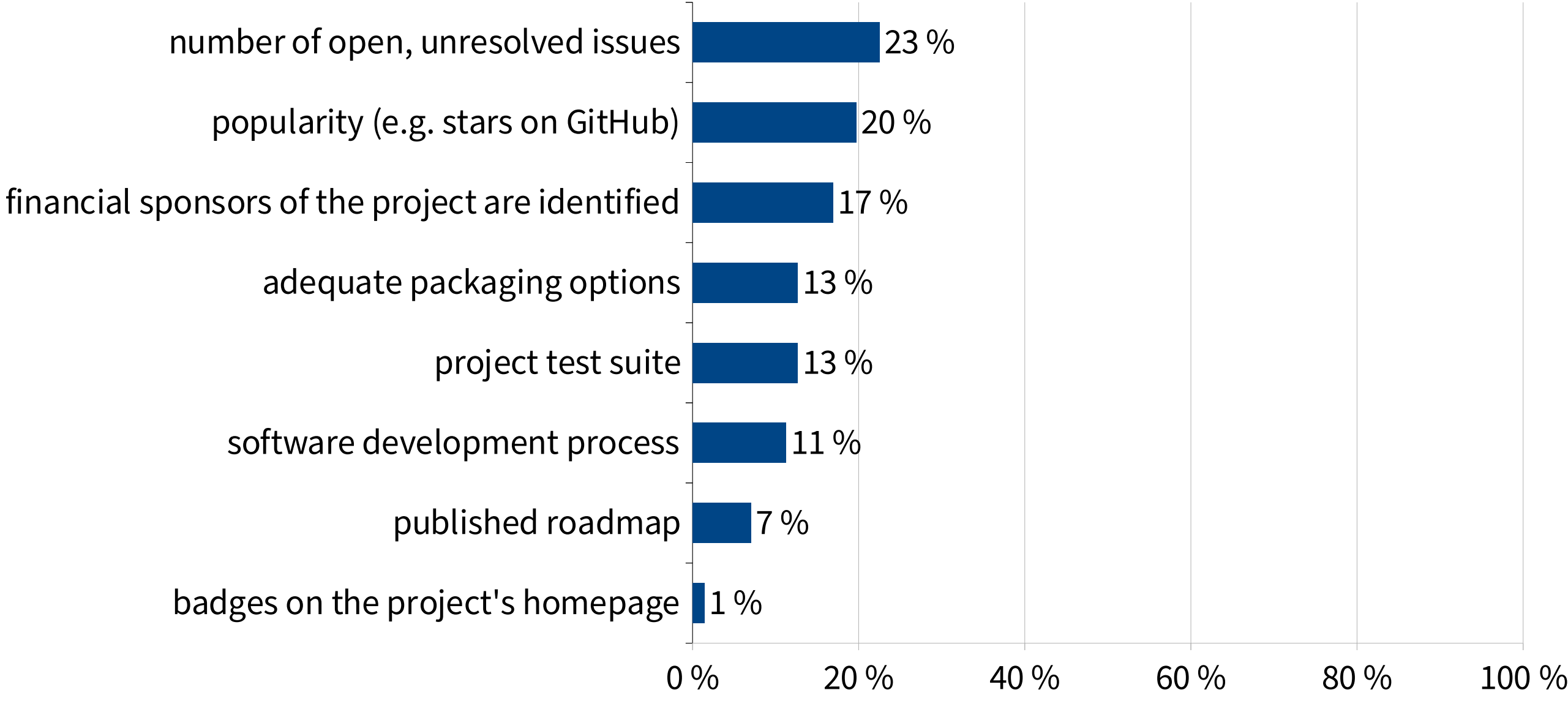
Mission critical software



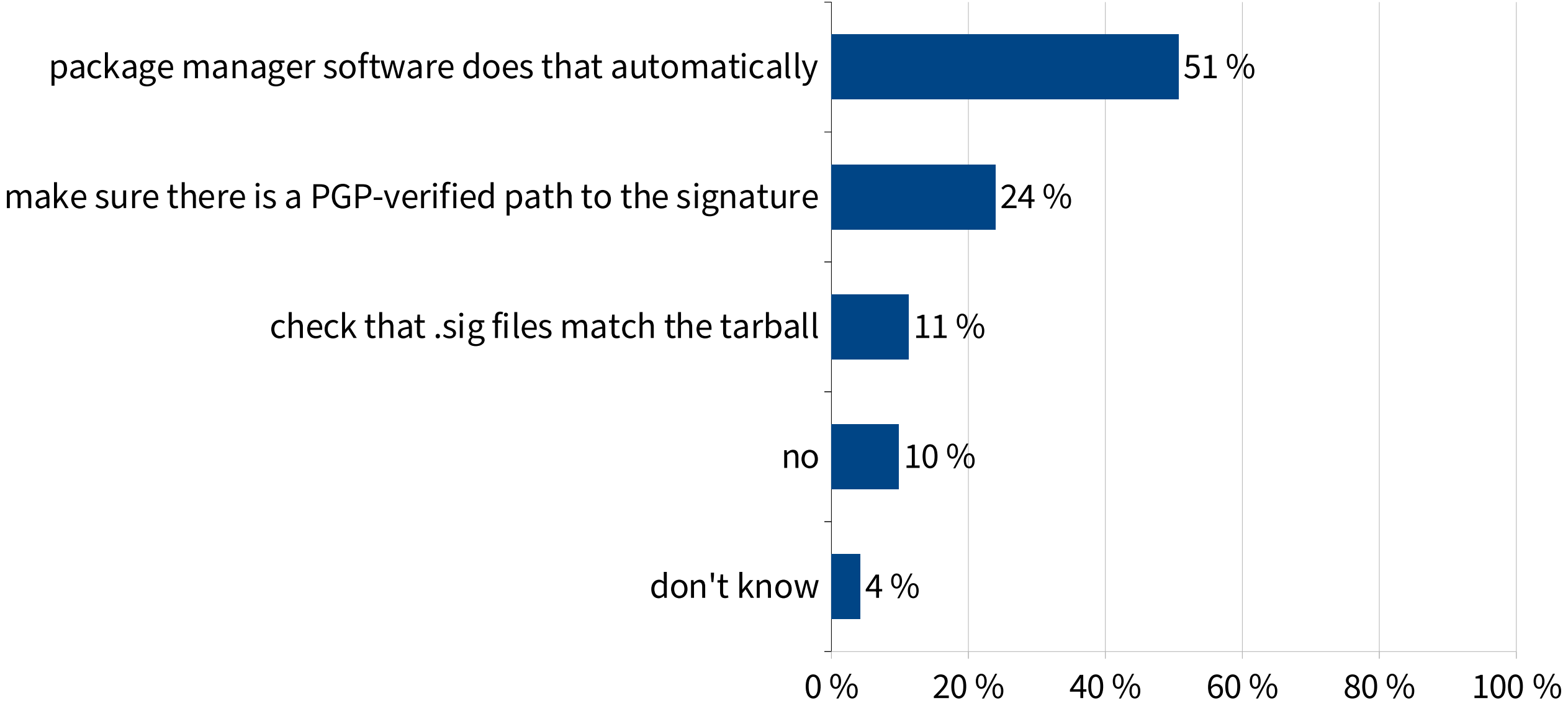
How do you build confidence? #1



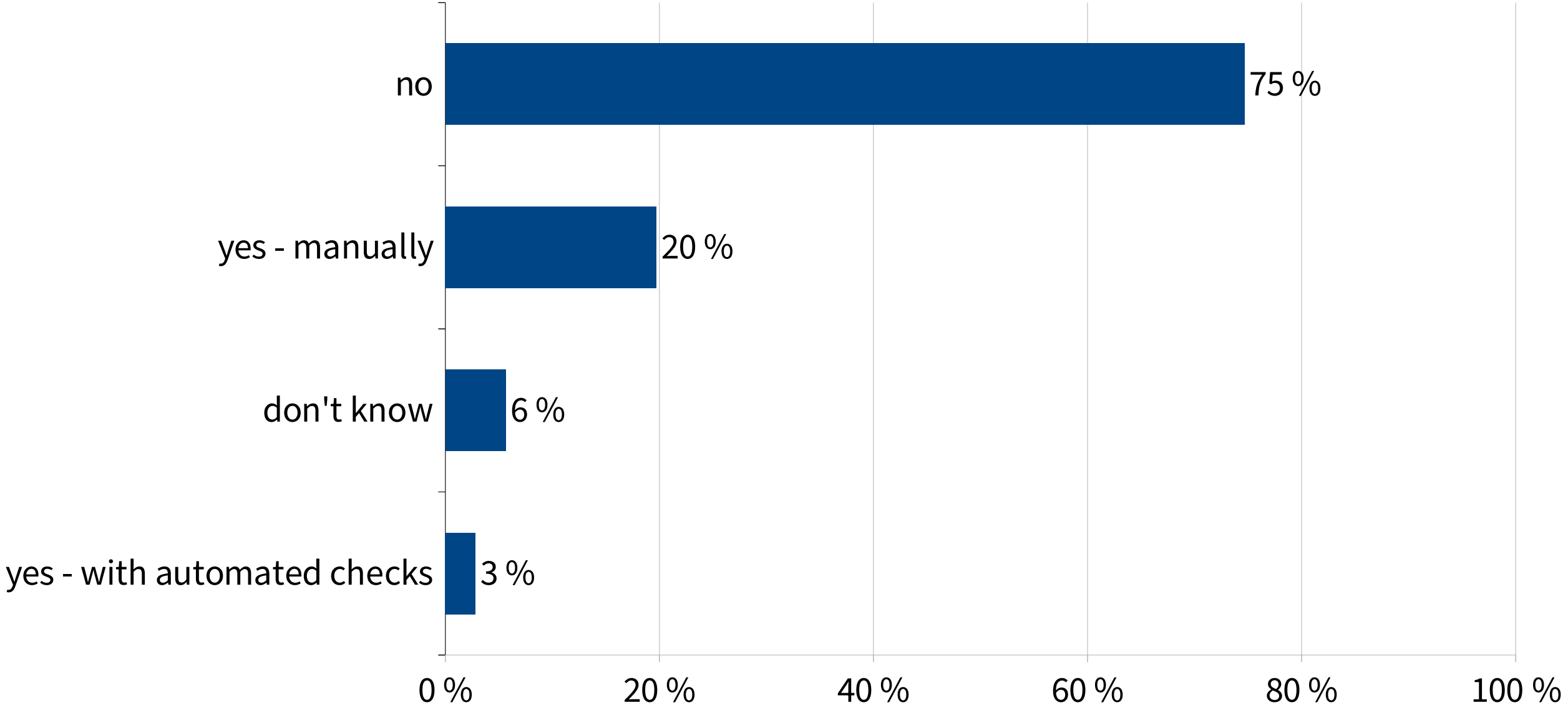
How do you build confidence? #2



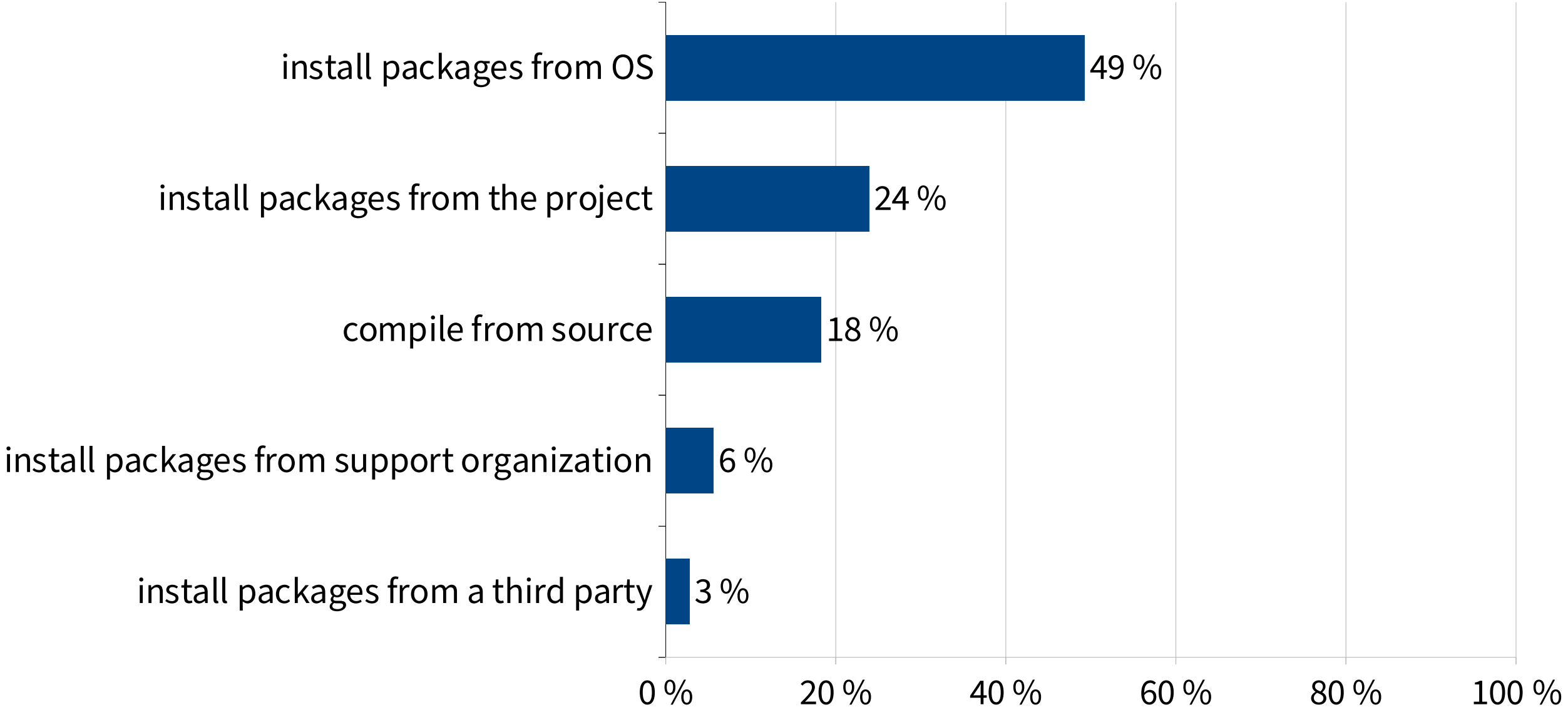
How do you verify signatures?



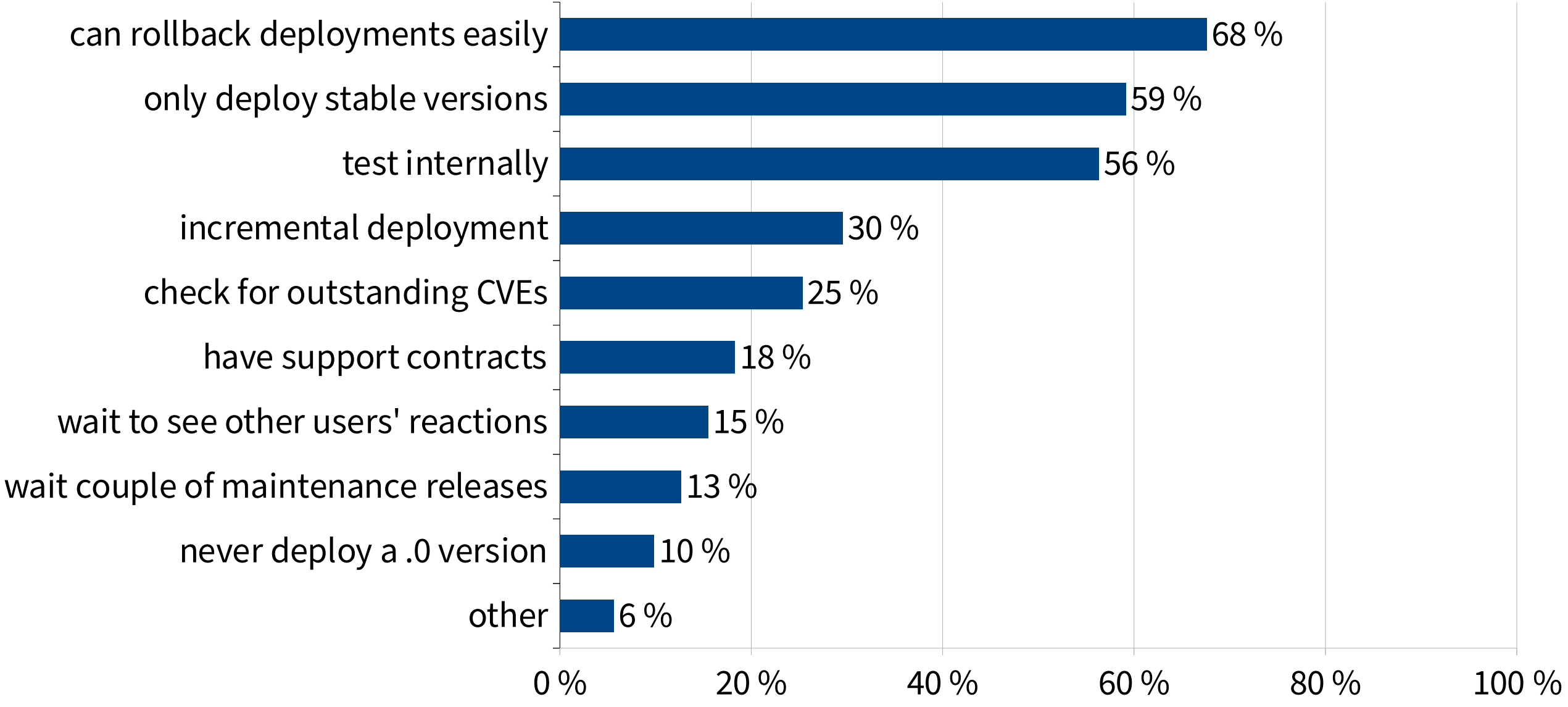
Do you inspect source code?



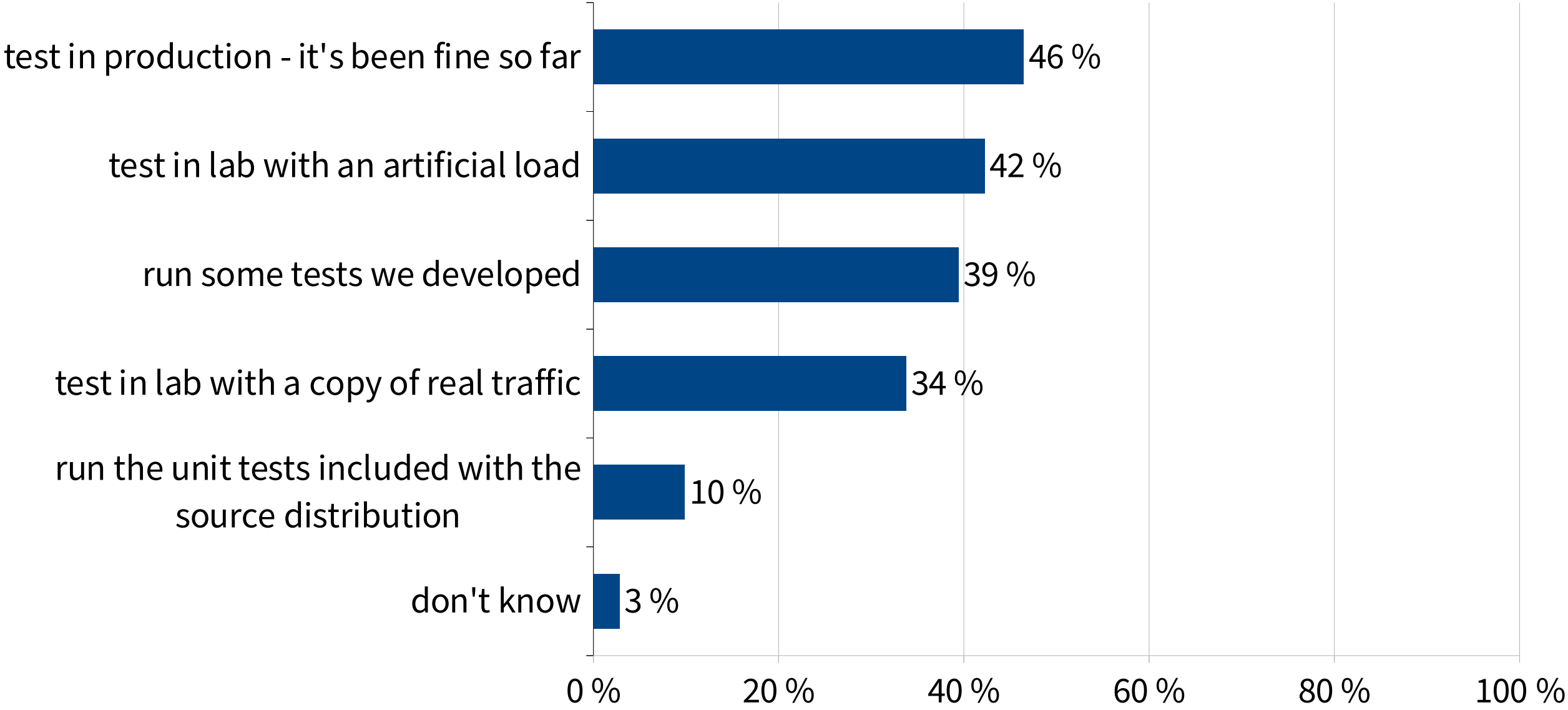
How do you install software?



How do you mitigate upgrade risks?



How do you test before production?



A DNS server – example project

BIND 9 in numbers

First commit	1998	26 years ago!
C code	263 000	lines (w/o tests or comments)
Automake	3 143	lines
Autoconf	1 839	lines
M4	1 626	lines
# of authors	50+	in the current codebase
# config knobs	325+	some are context dependent
# CVEs	130	mostly DoS

* attribution of old code is hard – squash & merge model

Existing code – audit

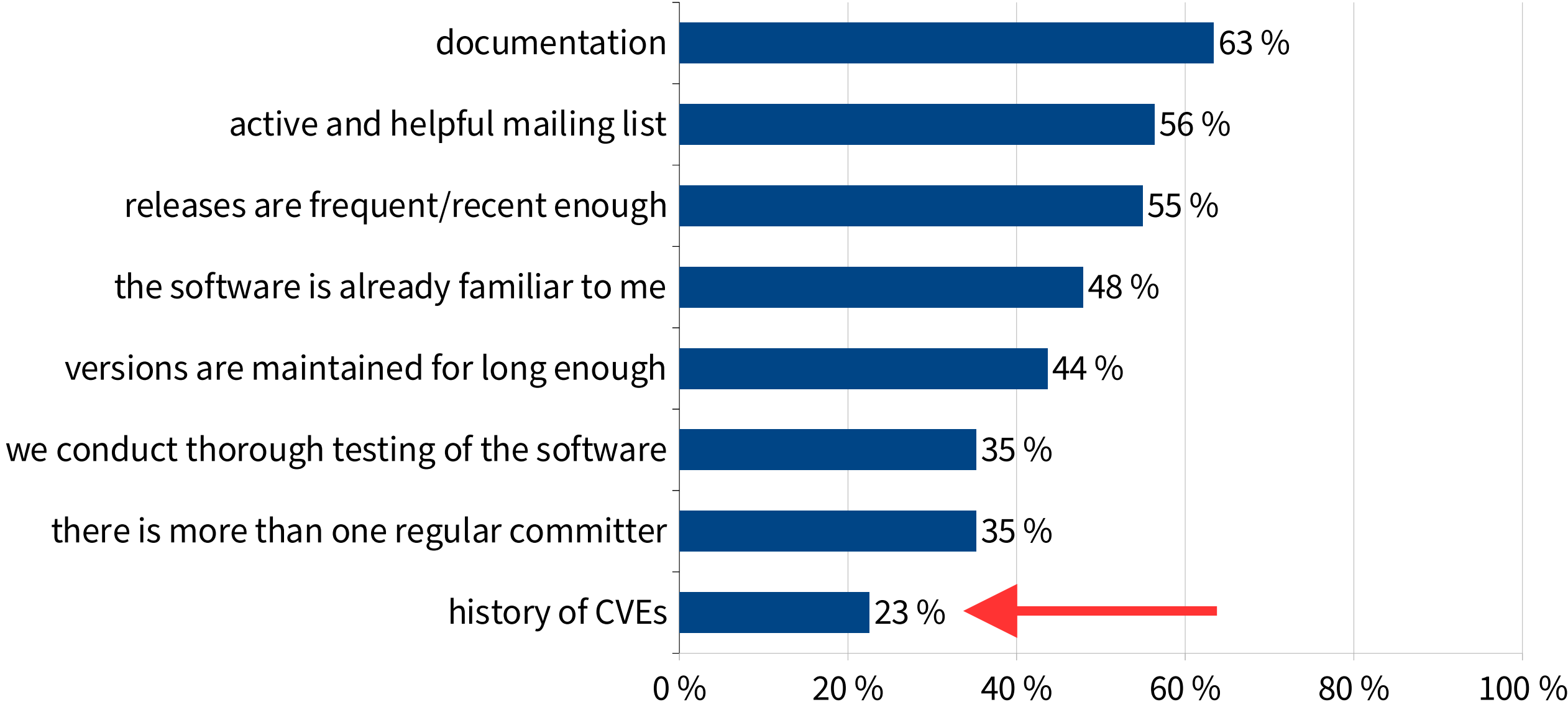
- Who knows what's in there?!
  - 26 years!
- Security audit in 2023
  - https://www.isc.org/blogs/2024-bind-audit/
  - 1 CVE, 2 medium severity, 6 low, 23 "nits" …
    - Low-level bugs

Audit limits

- No DNS-protocol level bugs found
  - By auditors – non-DNS experts
    - Meanwhile …

CVE #	Short Description
2023-50868	Preparing an NSEC3 closest encloser proof can exhaust CPU resources
2023-50387	KeyTrap - Extreme CPU consumption in DNSSEC validator
2023-6516	Specific recursive query patterns may lead to an out-of-memory condition
2023-5680	Cleaning an ECS-enabled cache may cause excessive CPU load
2023-5679	Enabling both DNS64 and serve-stale may cause an assertion failure …
2023-5517	Querying RFC 1918 reverse zones may cause an assertion failure when …
2023-4408	Parsing large DNS messages may cause excessive CPU load

BIND 9 vs. survey – CVEs

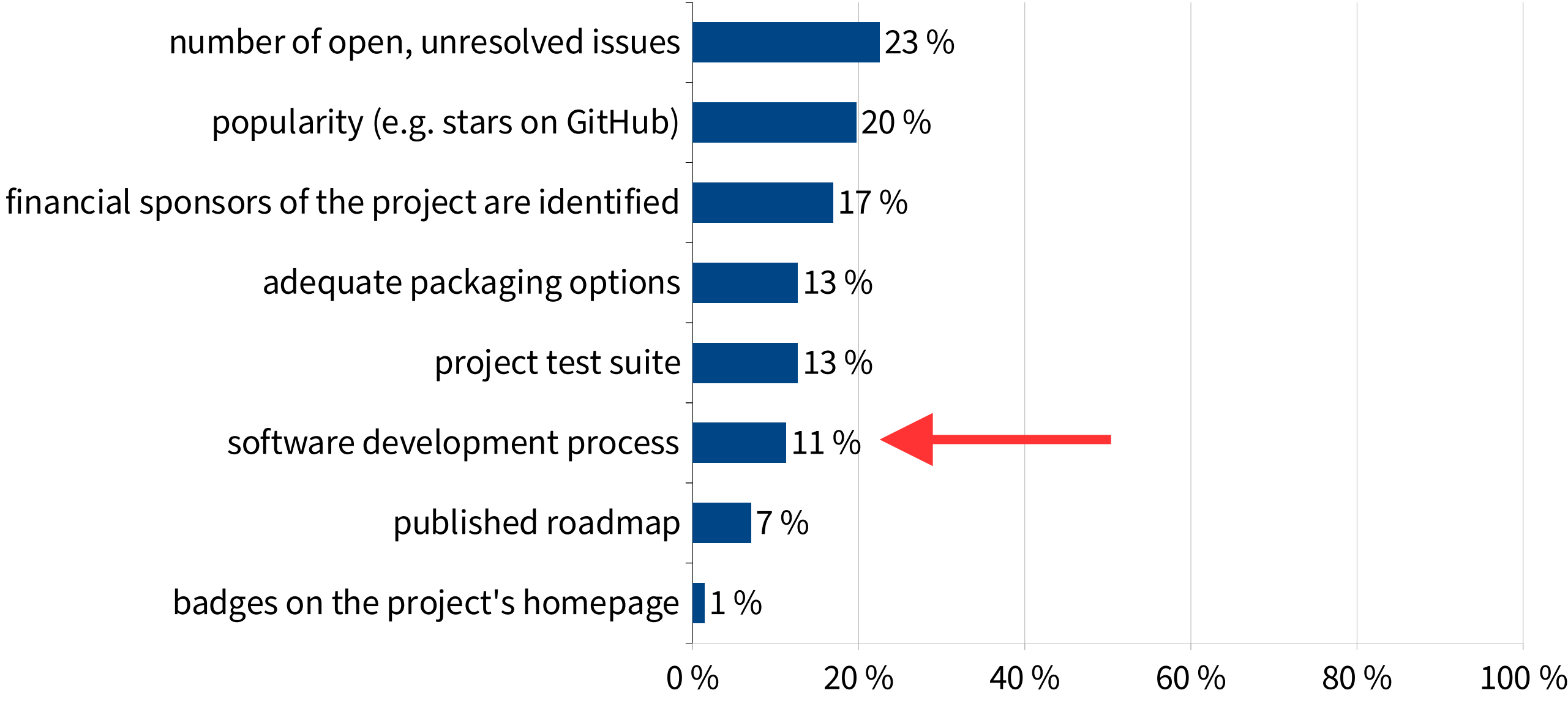


Self-imposed policies

- Coding & review procedures
- OpenSSF software quality badge 

  - Lots of non-technical requirements
- ISC software defect and security vulnerability disclosure
- ISC CVSS scoring guidelines
- A lot of invisible work

BIND 9 vs. survey – processes



BIND 9 new code

- Peer review in GitLab
  - Very few external contributions
- Automated tests
  - Continuous integration in GitLab
  - Extra things "on side"

Automated tests

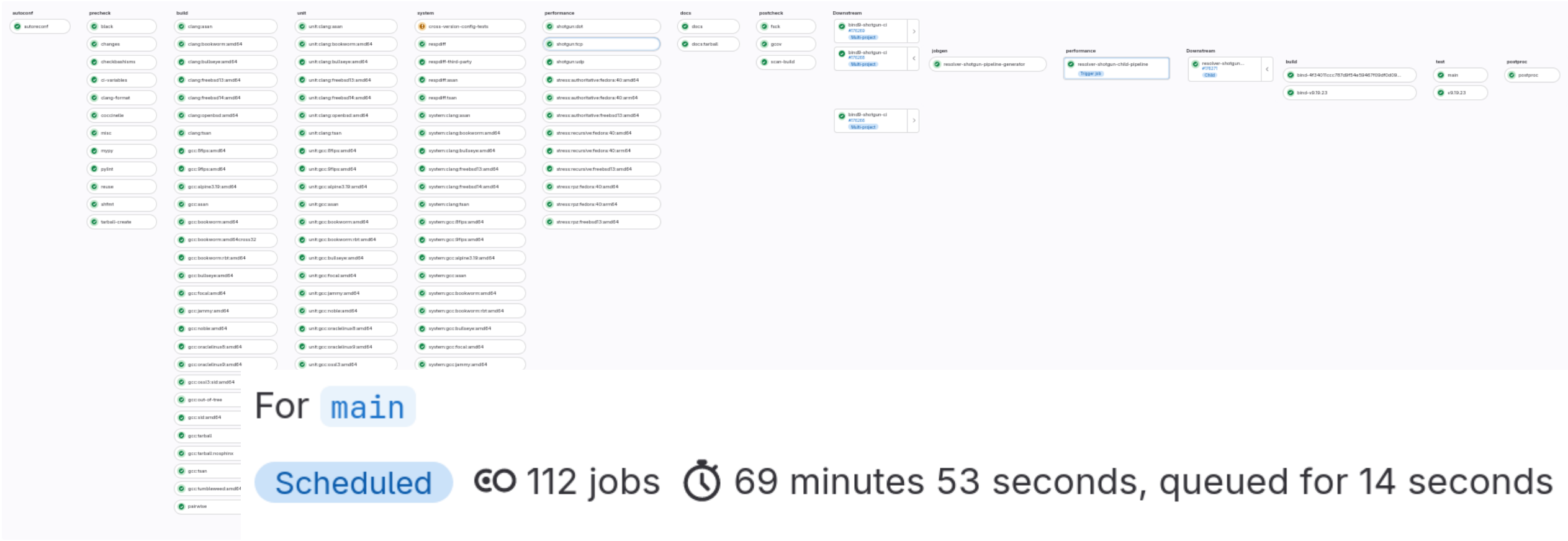
- Unit tests
- Integration
- Fuzzers
- Interoperability
- Stress
- Performance …

GCC Code Coverage Report

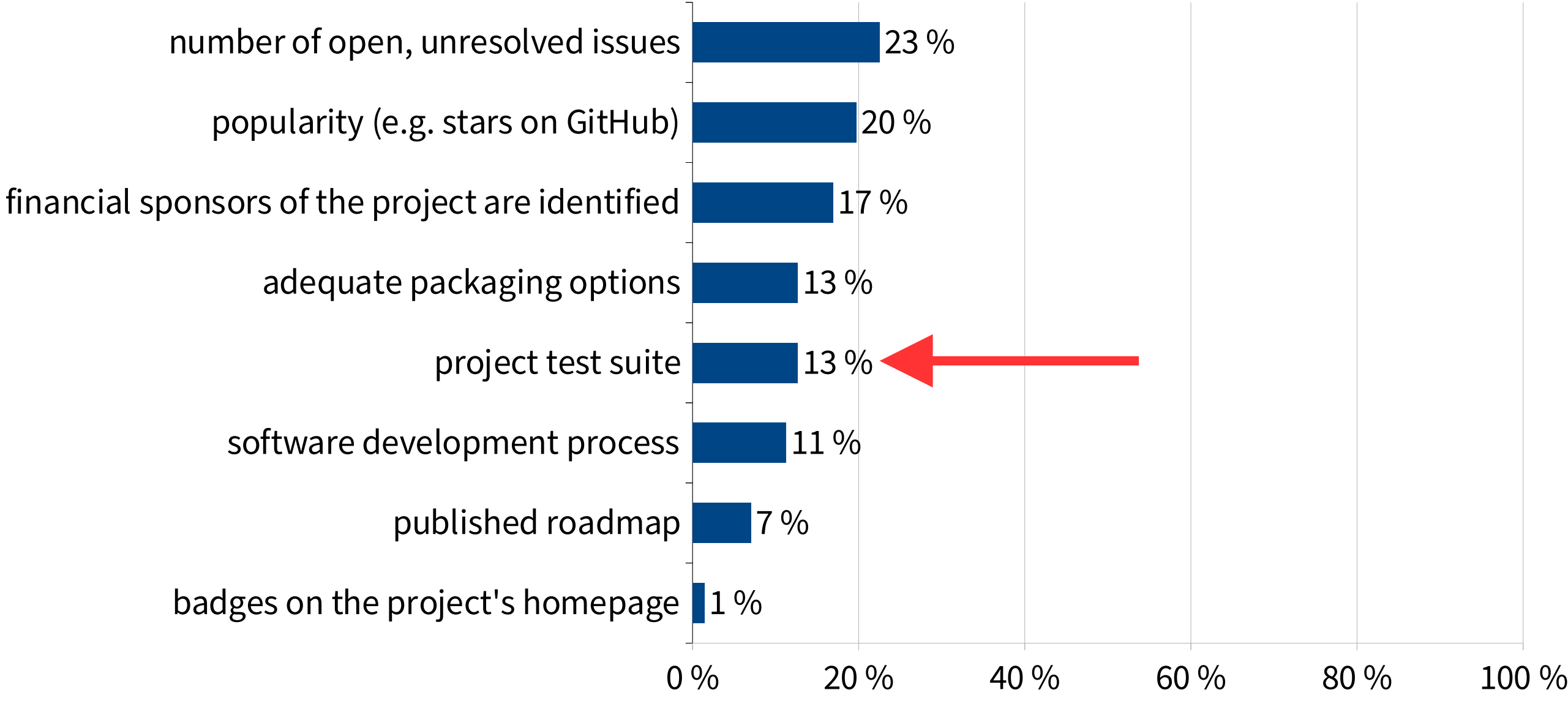
Metric	Coverage
Lines	77.1 %
Functions	85.5 %
Branches	55.6 %

incl. 12 204 assertions,
65.3 % without them

Continuous integration



BIND 9 vs. survey – tests



Peer review

Peer review

16 files +141 −383

ISC Open Source Projects / BIND / Merge requests / !5071

Merged **use a fixedname buffer in dns_message_gettempname()** 2515-improve-glue-cache-pe… into main

Overview 0 | Commits 2 | Pipelines 6 | **Changes 16**

Search (e.g. *.vue) (Ctrl+P)

- include/dns
  - message.h +3 -20
- win32
  - libdns.def.in +0 -1
- message.c +56 -107
- rbtdb.c +40 -129
- resolver.c +0 -1
- tkey.c +4 -18
- tsig.c +3 -6
- xfrin.c +0 -2
- zone.c +0 -3
- ns
  - client.c +12 -38
  - query.c +8 -21
  - xfrout.c +0 -2

lib/dns/rbtdb.c

```
9927      -          0xfc, 0xfd, 0xfe, 0xff
          9888  +  static const unsigned char maptoupper[256] = {
          9889  +          ['a'] = 'A', ['b'] = 'B', ['c'] = 'C', ['d'] = 'D', ['e'] = 'E',
          9890  +          ['f'] = 'F', ['g'] = 'G', ['h'] = 'H', ['i'] = 'I', ['j'] = 'J',
          9891  +          ['k'] = 'K', ['l'] = 'L', ['m'] = 'M', ['n'] = 'N', ['o'] = 'O',
          9892  +          ['p'] = 'P', ['q'] = 'Q', ['r'] = 'R', ['s'] = 'S', ['t'] = 'T',
          9893  +          ['u'] = 'U', ['v'] = 'V', ['x'] = 'X', ['y'] = 'Y', ['z'] = 'Z',
9928      9894     };
```

8 ✓ Approve

```
                                                  const dns_rdataset_t *rdataset, dns_name_t *name) {
                                                  te1;
                                                  rivate2;
                                                  te3; /* RDATASLAB */
9937      -          unsigned char bits;
9938      -          unsigned char c, flip;
          9901  +          rdatasetheader_t *header = NULL;
          9902  +          uint8_t mask = (1 << 7);
          9903  +          uint8_t bits = 0;
9939      9904
9940      9905           header = (struct rdatasetheader *)(raw - sizeof(*header));
9941      9906
          @@ -9946,85 +9911,36 @@ rdataset_getownercase(const dns_rdataset_t *rdataset, dns_name_t *name) {
```

Release … err, try again …

ISC (https://fosstodon.org/@iscdotorg)
@ISCdotORG

Ok, this is embarrassing. Please don't install the @bind9 updated versions we posted yesterday. Someone reported an error - we left out a LETTER of the ALPHABET in a streamlined routine. We will be removing the new versions and reposting after we have a chance to retest.

5:24 AM · Jun 18, 2021

Automated tests – limits

ISC Open Source Projects / BIND / Issues / #2779

W or w characters in domain names are altered to "\000"

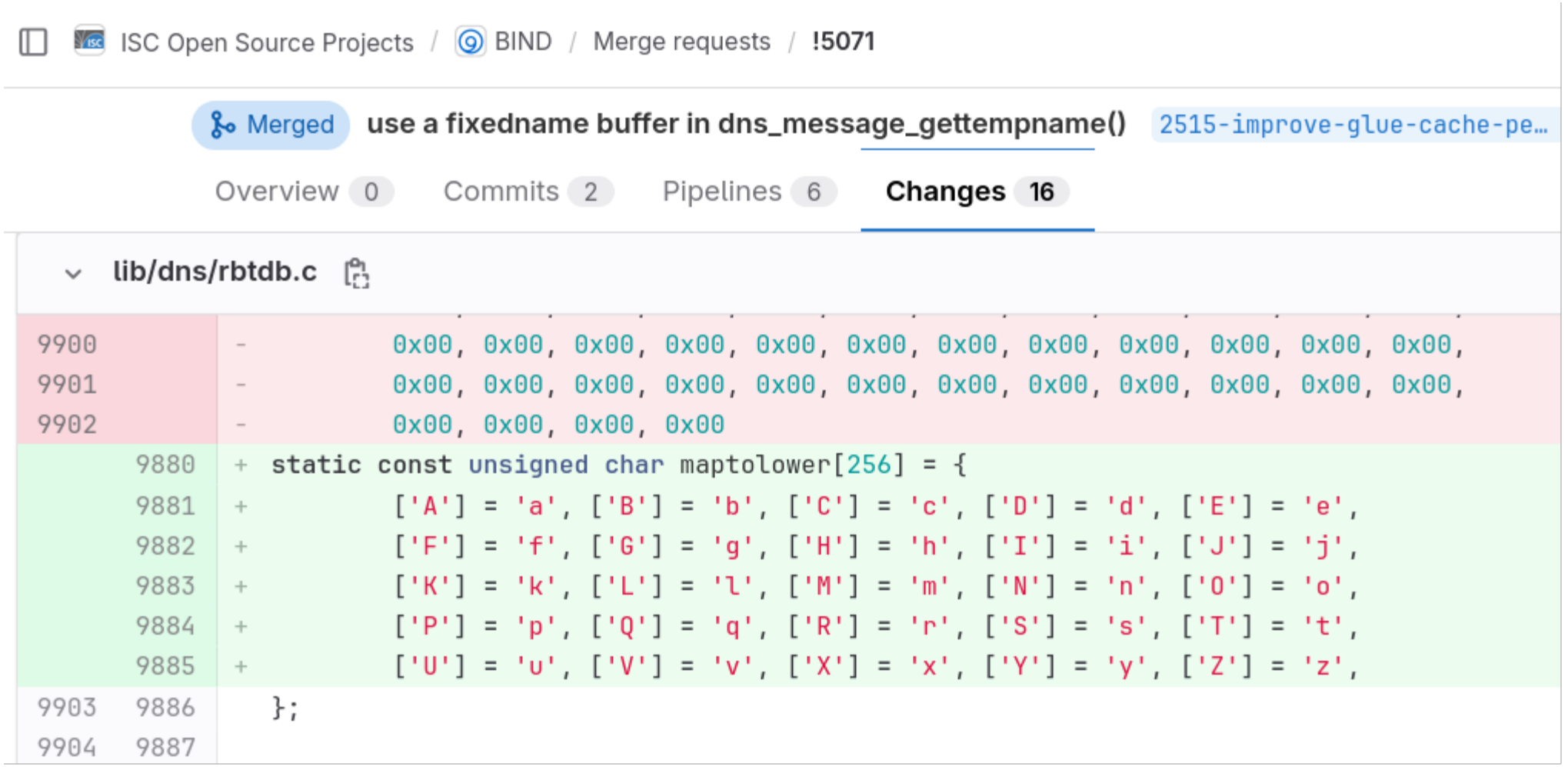
Closed Issue created 2 years ago by **Sean Zhang**

Summary

We recently upgraded our bind9 from `1:9.16.16-2+ubuntu18.04.1+isc+1` to `1:9.16.17-1+ubuntu21.04.1+isc+1` and start experiencing some wildcard names not being resolved. The resolver will return `servfail`. After some troubleshooting we found that:

Under certain conditions (reproducible), the name in answer will not match the name in question. Found this issue reproducible with following conditions:

Peer review – limits



The matrix is square! It must be fine!

BIND 9 Release process

Release Checklist

Before the Code Freeze

- [x] *(QA)* Rebase -S editions on top of current open-source versions: `git checkout bind-9.18-sub && git rebase origin/bind-9.18`
- [x] *(QA)* Inform Support and Marketing of impending release (and give estimated release dates).
- [x] *(QA)* Ensure there are no permanent test failures on any platform. Check public and private scheduled pipelines.
- [x] *(QA)* Check charts from `shotgun:*` jobs in the scheduled pipelines to verify there is no unexplained performance drop for any protocol.
- [x] *(QA)* Check Perflab to ensure there has been no unexplained drop in performance for the versions being released.
- [x] *(QA)* Check whether all issues assigned to the release milestone are resolved[1].
- [x] *(QA)* Ensure that there are no outstanding merge requests in the private repository[1] (Subscription Edition only).
- [x] *(QA)* Ensure all merge requests marked for backporting have been indeed backported.
- [x] *(QA)* Announce (on Mattermost) that the code freeze is in effect.

Before the Tagging Deadline

- [x] *(QA)* Inspect the current output of the `cross-version-config-tests` job to verify that no unexpected backward-incompatible change was introduced in the current release cycle.
- [x] *(QA)* Ensure release notes are correct, ask Support and Marketing to check them as well. Example
- [x] *(QA)* Add a release marker to `CHANGES`. Examples: `9.18`, `9.16`
- [x] *(QA)* Add a release marker to `CHANGES.SE` (Subscription Edition only). Example
- [x] *(QA)* Update BIND 9 version in `configure.ac` (`9.18+`) or `version` (`9.16`).
- [ ] 🚫 *(QA)* Rebuild `configure` using Autoconf on `docs.isc.org` (9.16).
- [ ] 🚫 *(QA)* Update GitLab settings for all maintained branches to disallow merging to them: public, private
- [x] *(QA)* Tag the releases in the private repository (`git tag -s -m "BIND 9.x.y" v9.x.y`).

Before the ASN Deadline (for ASN Releases) or the Public Release Date (for Regular Releases)

- [x] *(QA)* Check that the formatting is correct for the HTML version of release notes.
- [x] *(QA)* Check that the formatting of the generated man pages is correct.
- [x] *(QA)* Verify GitLab CI results for the tags created and sign off on the releases to be published.
- [ ] 🚫 *(QA)* Update GitLab settings for all maintained branches to allow merging to them again: public, private
- [x] *(QA)* Prepare (using `version_bump.py`) and merge MRs resetting the release notes and updating the version string for each maintained branch.
- [x] *(QA)* Rebase the Subscription Edition branches (including recent release prep commits) on top of the open source branches with updated version strings.
- [x] *(QA)* Announce (on Mattermost) that the code freeze is over.
- [x] *(QA)* Request signatures for the tarballs, providing their location and checksums. Ask signers on Mattermost.
- [x] *(Signers)* Ensure that the contents of tarballs and tags are identical.
- [x] *(Signers)* Validate tarball checksums, sign tarballs, and upload signatures.
- [x] *(QA)* Verify tarball signatures and check tarball checksums again: Run `publish_bind.sh` on repo.isc.org to pre-publish.
- [ ] 🚫 *(QA)* Prepare the `patches/` subdirectory for each security release (if applicable).
- [x] *(QA)* Pre-publish ASN and/or Subscription Edition tarballs so that packages can be built.
- [x] *(QA)* Build and test ASN and/or Subscription Edition packages (in cloudsmith branch in private repo). Example
- [ ] 🚫 *(Marketing)* Prepare and send out ASN emails (as outlined in the CVE checklist; if applicable).

On the Day of Public Release

- [ ] 🚫 *(QA)* Wait for clearance from Security Officer to proceed with the public release (if applicable).
- [x] *(QA)* Place tarballs in public location on FTP site.
- [x] *(QA)* Inform Marketing of the release, providing FTP links for the published tarballs.
- [x] *(QA)* Use the Printing Press project to prepare a release announcement email.
- [x] *(Marketing)* Publish links to downloads on ISC website. Example
- [x] *(Marketing)* Update the BIND -S information document in SF with download links to the new versions. (If this is a security release, this will have already been done as part of the ASN process.)
- [x] *(Marketing)* Update the Current Software Versions document in the SF portal if any stable versions were released.
- [x] *(Marketing)* Send the release announcement email to the *bind-announce* mailing list (and to *bind-users* if a major release - example).
- [x] *(Marketing)* Announce release on social media sites.
- [x] *(Marketing)* Update Wikipedia entry for BIND.
- [x] *(Support)* Add the new releases to the vulnerability matrix in the Knowledge Base.
- [x] *(Support)* Update tickets in case of waiting support customers.
- [x] *(QA)* Build and test any outstanding private packages in private repo. Example
- [x] *(QA)* Build public RPMs. Example `commit` which triggers Copr builds automatically
- [x] *(SwEng)* Build Debian/Ubuntu packages.
- [x] *(SwEng)* Update Docker files here and make sure push is synchronized to GitHub. Docker Hub should pick it up automatically. Example
- [x] *(QA)* Ensure all new tags are annotated and signed. `git show --show-signature v9.19.12`
- [x] *(QA)* Push tags for the published releases to the public repository.
- [x] *(QA)* Using `merge_tag.py`, merge published release tags back into the their relevant development/maintenance branches.
- [x] *(QA)* Ensure `allow_failure: true` is removed from the `cross-version-config-tests` job if it was set during the current release cycle.
- [x] *(QA)* Sanitize confidential issues which are assigned to the current release milestone and do not describe a security vulnerability, then make them public.
- [ ] 🚫 *(QA)* Sanitize confidential issues which are assigned to older release milestones and describe security vulnerabilities, then make them public if appropriate[2].
- [x] *(QA)* Update QA tools used in GitLab CI (e.g. Black, PyLint, Sphinx) by modifying the relevant `Dockerfile`.
- [x] *(QA)* Run a pipeline to rebuild all images used in GitLab CI.
- [x] *(QA)* Update `metadata.json` with the upcoming release information.

BIND 9 release process

- Check list of changes that went in (again)
- Polish docs
- Run tests (again)
- Generate tarball
  - Check reproducibility
- Sign
- Publish
- Build packages

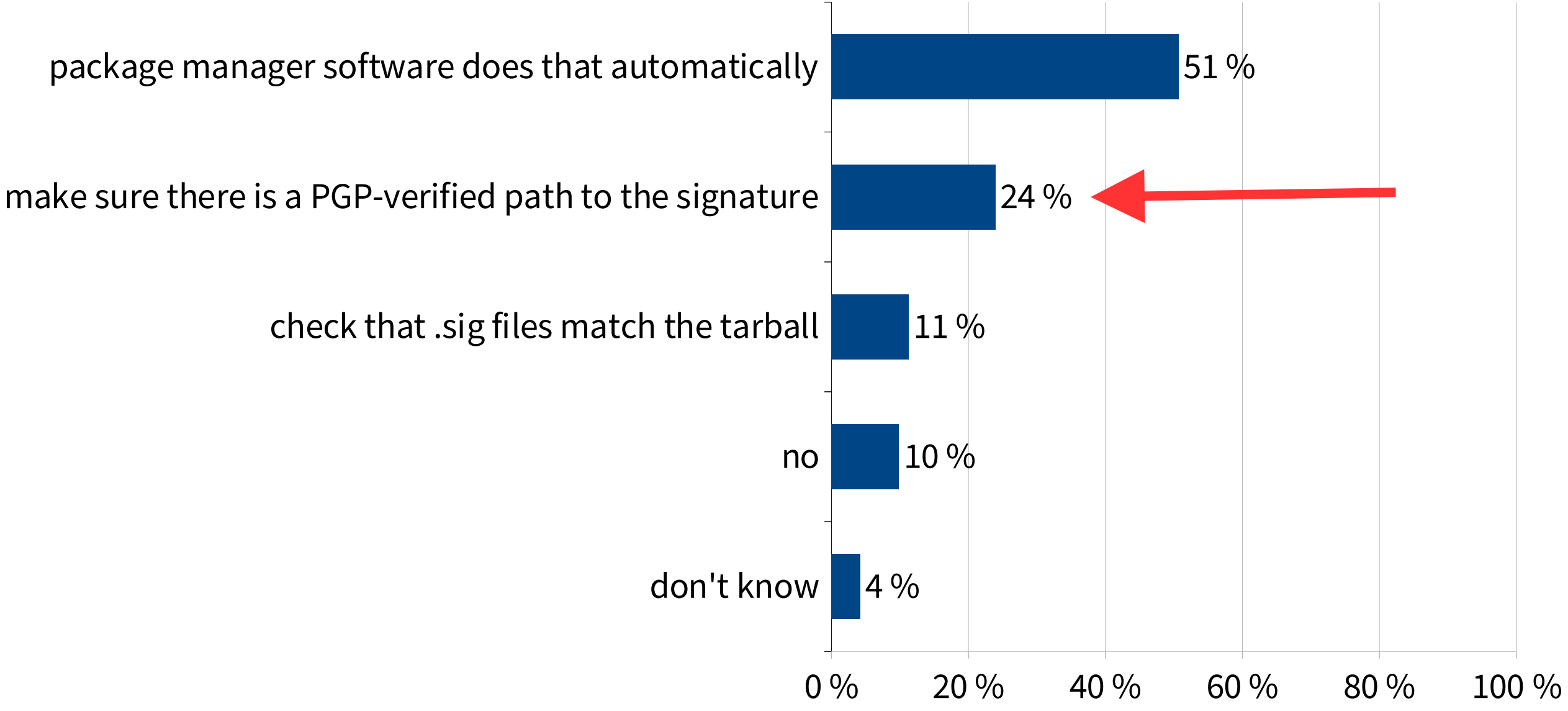
BIND 9 tarball checks

- Git ⇒ tarball reproducibility
  - https://gitlab.isc.org/isc-projects/BIND9/-/blob/main/util/release-tarball-comparison.sh
  - 100 lines
  - easy enough for independent review

BIND 9 tarball signing

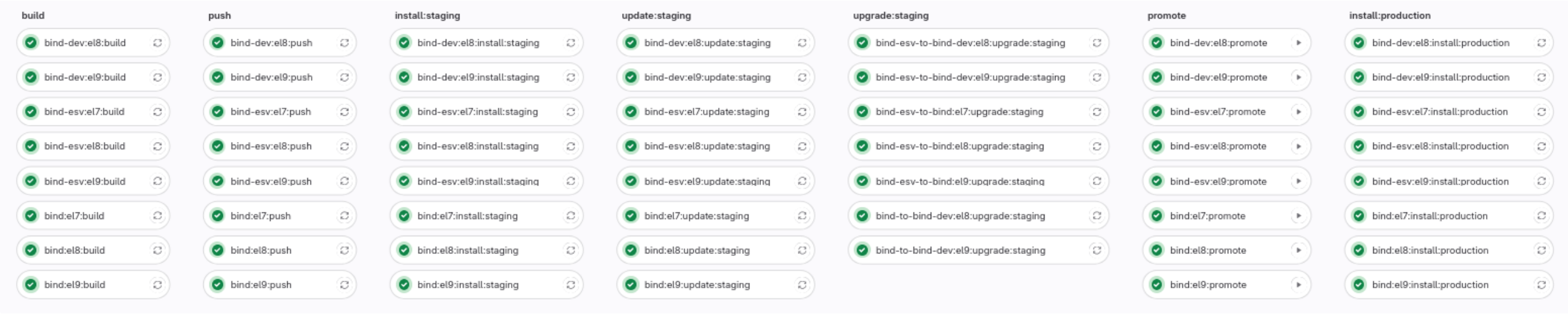
- Dedicated VM
  - takes tarball from Gitlab
  - requests GPG signature
- Signer – person
  - SSH into the VM
  - forwards GPG agent socket

BIND 9 vs. survey – signatures

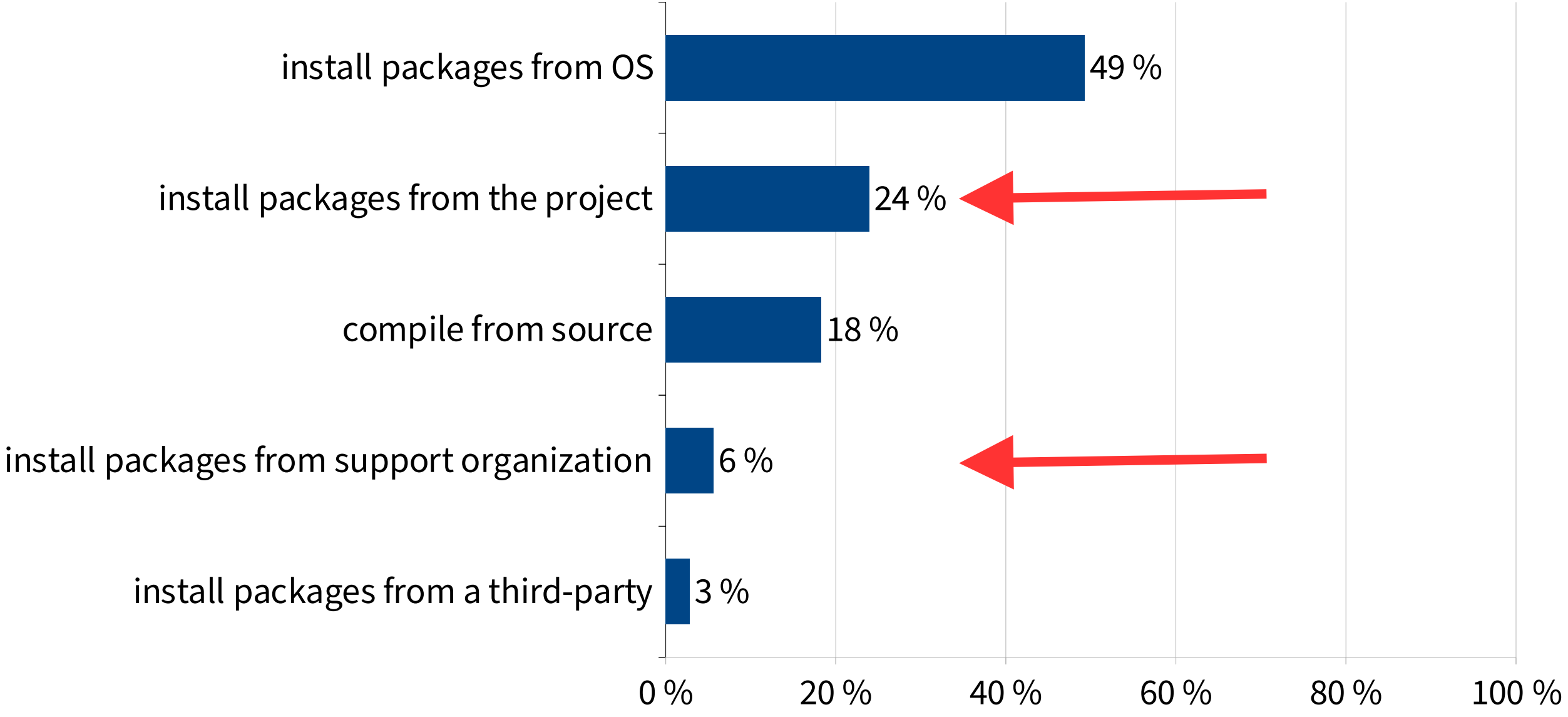


BIND 9 package build

- Our RPM packages build in Gitlab
- Copr, Launchpad, Docker, etc. – manual



BIND 9 vs. survey – packages



BIND 9 team vs. survey

	Team priority	Survey priority
CVE frequency	#1	# 8
CI & automated tests	#1	# 11
code reviews & standards	#1	# 13

Discussion

Thank you!

- Main website: https://www.isc.org
- Software downloads: https://www.isc.org/download or https://downloads.isc.org
- Presentations: https://www.isc.org/presentations
- Main GitLab: https://gitlab.isc.org