Notary V2 Overview

Author: Nho Luong

Skill: DevOps Engineer Lead

Microsoft

Nho Luong

has successfully passed all requirements for

Microsoft Certified: Azure Solutions Architect Expert

Credential ID: 3CF3D14D253FE551
Certification number: 46F8FB-7D3F5B
Earned on: June 14, 2024
Expires on: June 15, 2025

Satya Narayana Nadella

✓ Online Verifiable

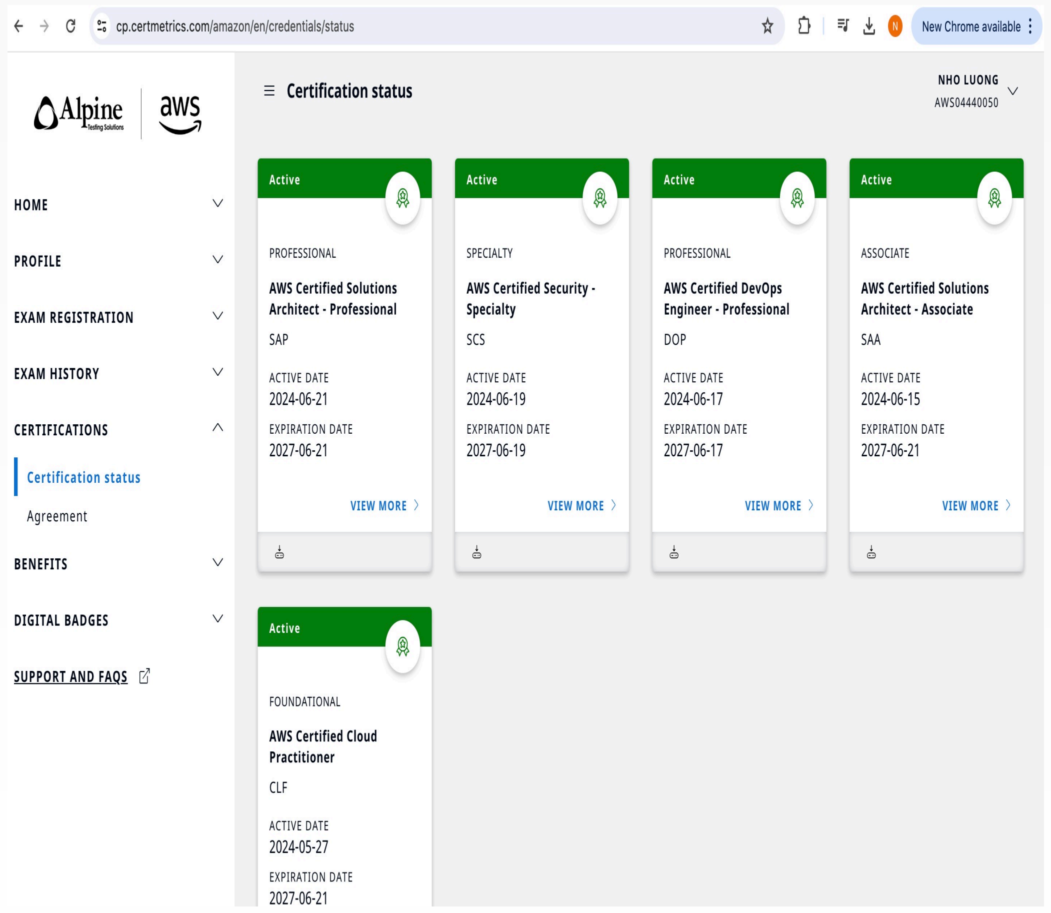
aws certified

nho luong

AWS Certified Solutions Architect - Professional

VALIDATION NUMBER: 119ee670510e4bb99fe2956152effb7e
VALIDATE AT: https://aws.amazon.com/verification

Issue Date: June 21, 2024
Expiration Date: June 21, 2027

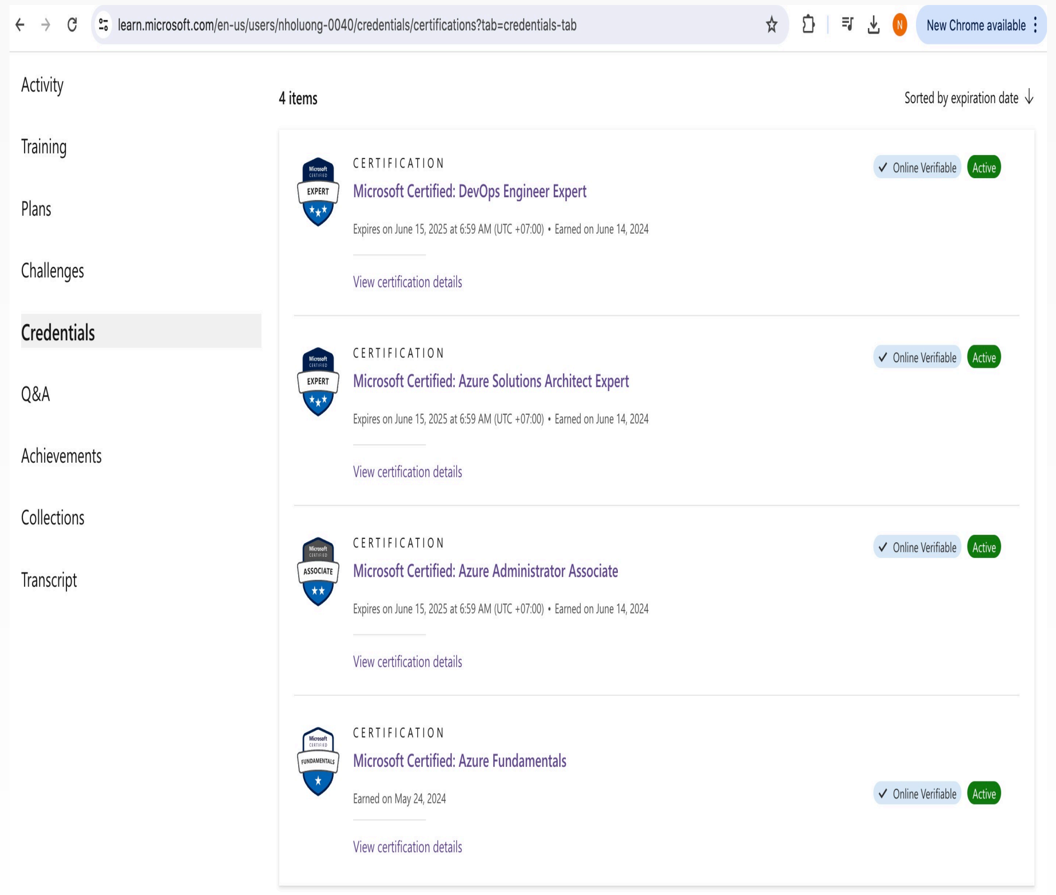

cp.certmetrics.com/amazon/en/credentials/status

Certification status

NHO LUONG
AWS04440050

HOME
PROFILE
EXAM REGISTRATION
EXAM HISTORY
CERTIFICATIONS
Certification status
Agreement
BENEFITS
DIGITAL BADGES
SUPPORT AND FAQS

Status	Level	Certification	Code	Active date	Expiration date
Active	PROFESSIONAL	AWS Certified Solutions Architect - Professional	SAP	2024-06-21	2027-06-21
Active	SPECIALTY	AWS Certified Security - Specialty	SCS	2024-06-19	2027-06-19
Active	PROFESSIONAL	AWS Certified DevOps Engineer - Professional	DOP	2024-06-17	2027-06-17
Active	ASSOCIATE	AWS Certified Solutions Architect - Associate	SAA	2024-06-15	2027-06-21
Active	FOUNDATIONAL	AWS Certified Cloud Practitioner	CLF	2024-05-27	2027-06-21


learn.microsoft.com/en-us/users/nholuong-0040/credentials/certifications?tab=credentials-tab

Activity
Training
Plans
Challenges
Credentials
Q&A
Achievements
Collections
Transcript

4 items — Sorted by expiration date

CERTIFICATION
Microsoft Certified: DevOps Engineer Expert
Expires on June 15, 2025 at 6:59 AM (UTC +07:00) • Earned on June 14, 2024
✓ Online Verifiable — Active

CERTIFICATION
Microsoft Certified: Azure Solutions Architect Expert
Expires on June 15, 2025 at 6:59 AM (UTC +07:00) • Earned on June 14, 2024
✓ Online Verifiable — Active

CERTIFICATION
Microsoft Certified: Azure Administrator Associate
Expires on June 15, 2025 at 6:59 AM (UTC +07:00) • Earned on June 14, 2024
✓ Online Verifiable — Active

CERTIFICATION
Microsoft Certified: Azure Fundamentals
Earned on May 24, 2024
✓ Online Verifiable — Active

What: is Notary v2

- Registry-native

Signatures and artifacts co-located for easier and secure management

- Secure

Attesting to its authenticity and/or certification
No trust on first use, no implicit permissions on rotated keys, secure private keys and PKI

- Portable

Artifacts move within and across registries supporting provenance, validation and trust

- Multi-tenant

Enable cloud providers and enterprises to easily support managed services at scale

- Offline & Air-gapped

Artifacts can be signed offline
Artifacts and signatures can be moved into air-gapped environments

- Usable

Simple commands to integrate with toolchains, supporting key hierarchies

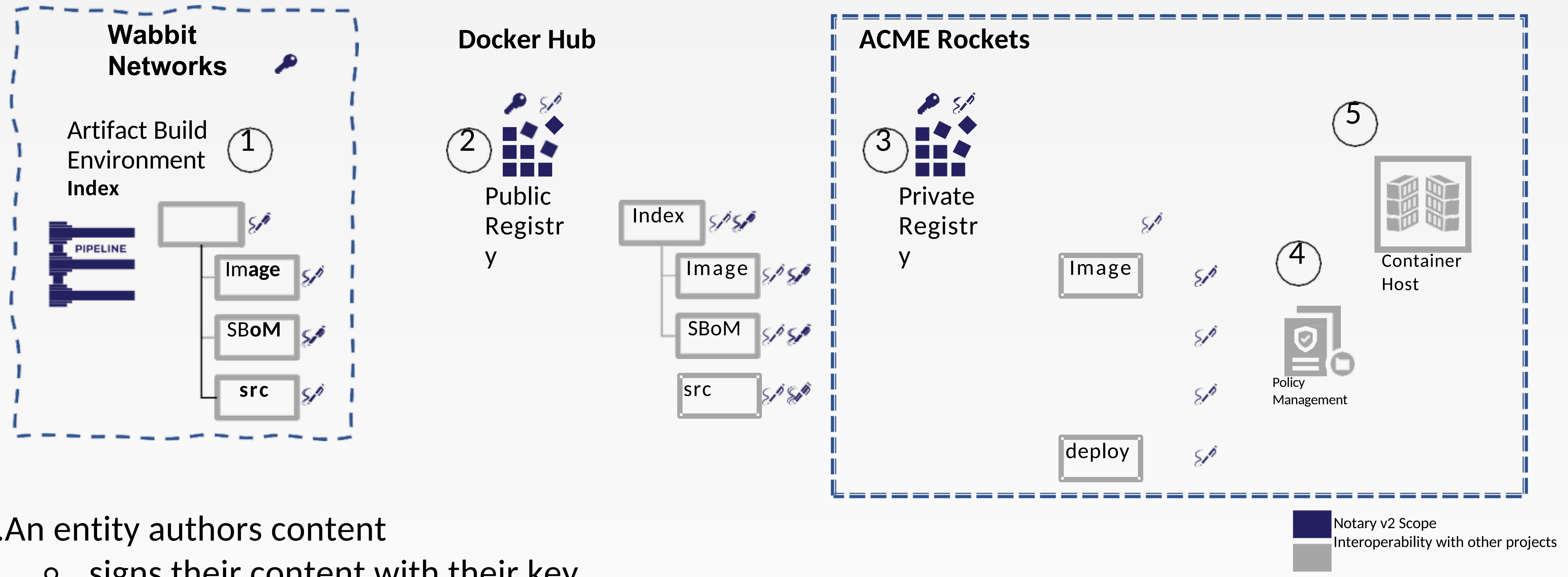
Notary v1 does not meet these requirements Notary v2 intends to

Notary v2 Requirements

1. Offline signing
2. Must not change the tag or digest, just to be signed
3. Cross cloud, on-prem and air-gapped adoption
4. Ephemeral clients
5. Multiple signatures
   - Enabling originating vendor, aggregator certification, customer validation
6. Keys secured by cloud providers key vault offering (pluggable)
7. Key acquisition: from hobbyist, open source projects, to large software vendors

Author: Nho Luong
Skill: DevOps Engineer Lead

Notary v2 Workflow

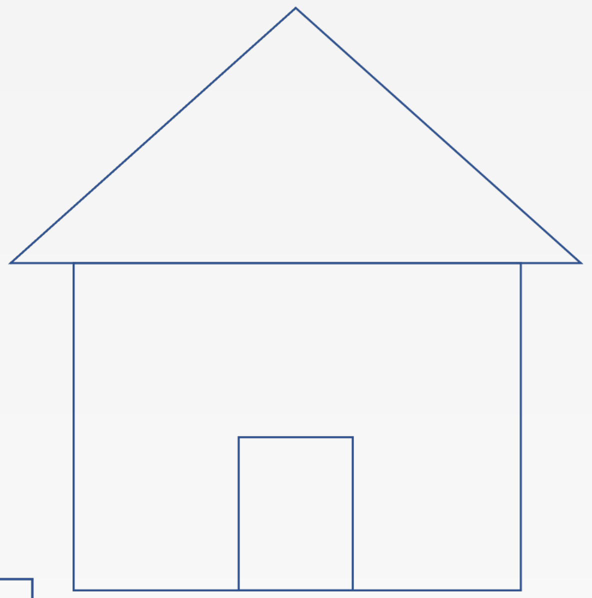


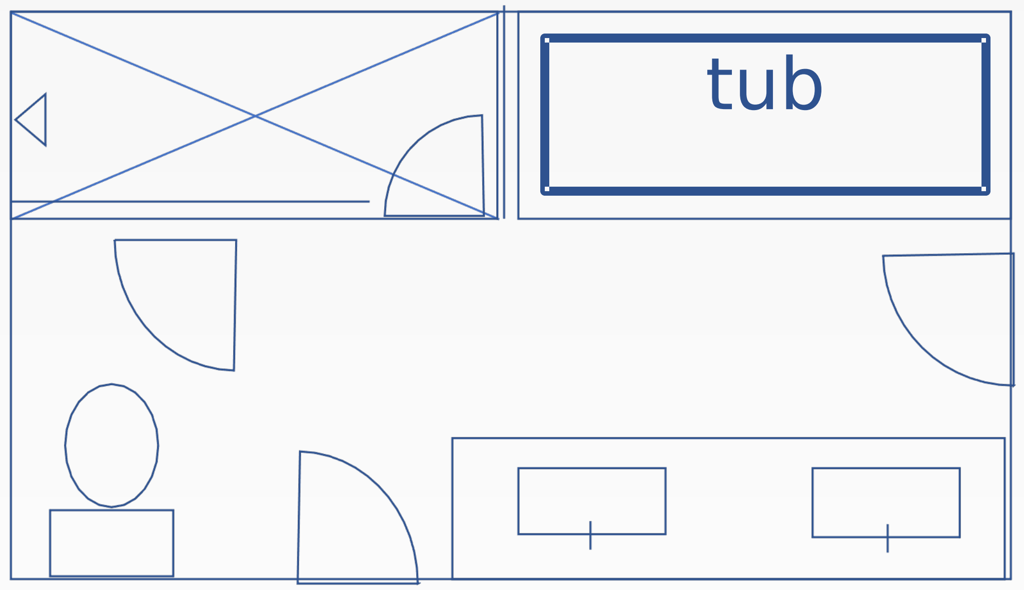
1. An entity authors content
    - signs their content with their key
2. Publish to a well-known location
    - May get certified by the aggregator
1. Consume the public content into an entity's private registry
    - Add a verification signature, attesting to its usage in the company
2. Policy management enforces which keys can be used for deployment, even what registries content can be pulled from
3. Only after all signatures and policies are verified can the artifact be deployed

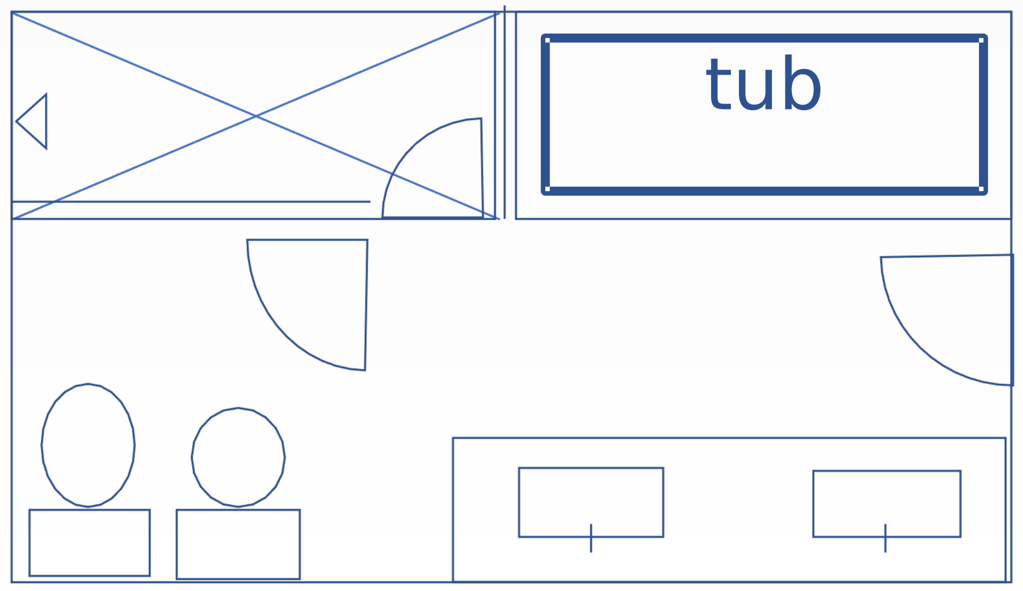
Author: Nho Luong
Skill: DevOps Engineer Lead

Prototyping Approach

- How to build complex systems?
  - How do we establish a model for communication?
- We want to build a house?
  - What does that mean?
  - What style?
  - How many rooms?
  - City, Suburb, Mountain, Beach?
  - What style of kitchen?
  - What style of bathroom?
- Enlisting expertise of the trades
  - Grading contractors
  - Foundation contractors
  - Framing contractors
  - HVAC contractors
  - Plumbing contractors
  - Electrical contractors



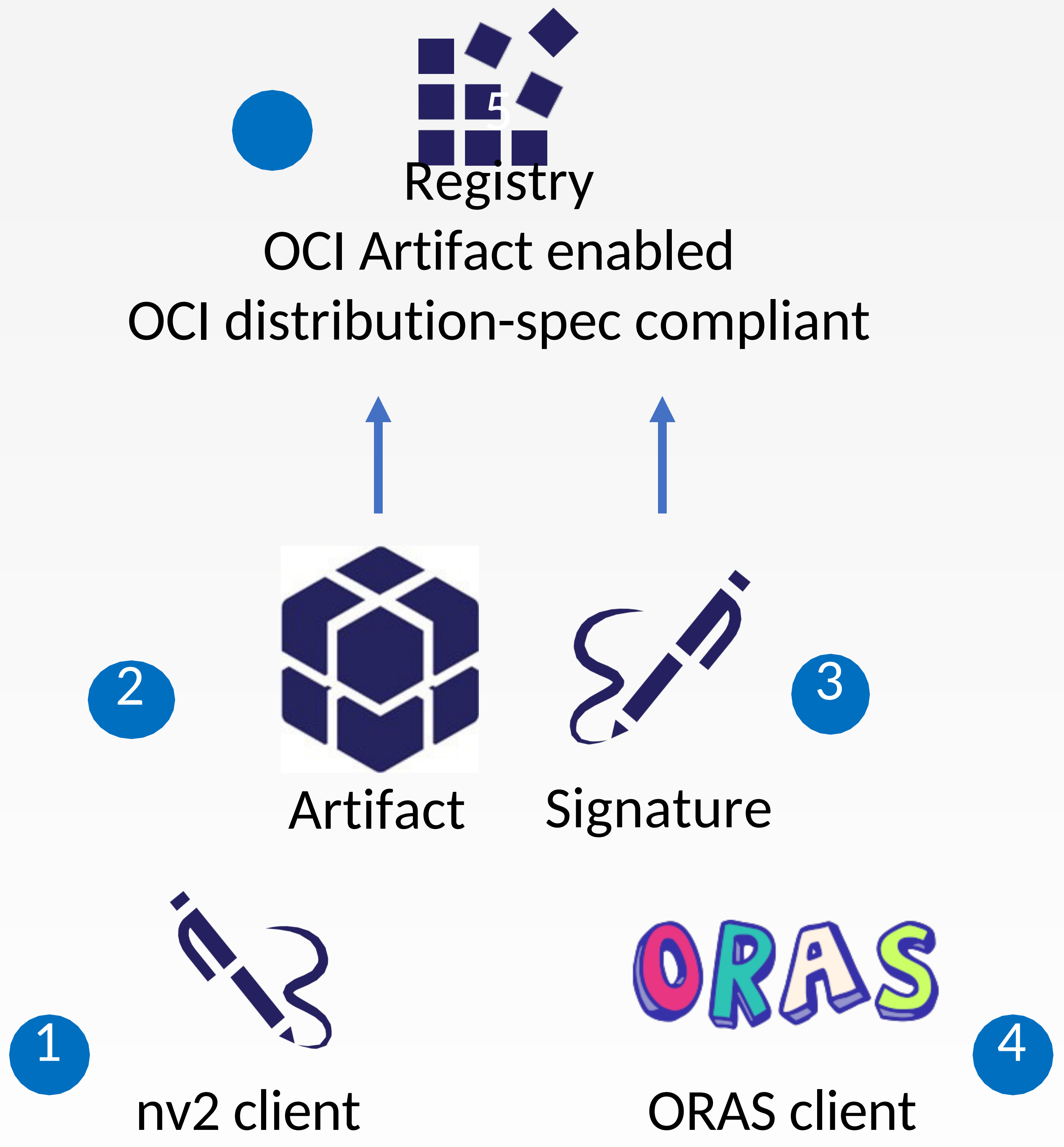




Where are we now?

- Prototyping to get closer to where we want to be
- Prototype 1
  - Generic signing of content
    - Supporting any content pushed to an OCI Artifacts enabled registry
    - Attesting to its authenticity and/or certification
  - Content copying, with signatures
    - within and across registries
    - Into air-gapped environments
  - Looking at the key management issues, types of keys
  - Registry persistence and retrieval
    - An artifact?
    - Different permissions?
- Further prototypes and design decisions
  - TUF
  - Rollback protection in a registry context
  - ephemeral clients and their issues

Breaking down the pieces

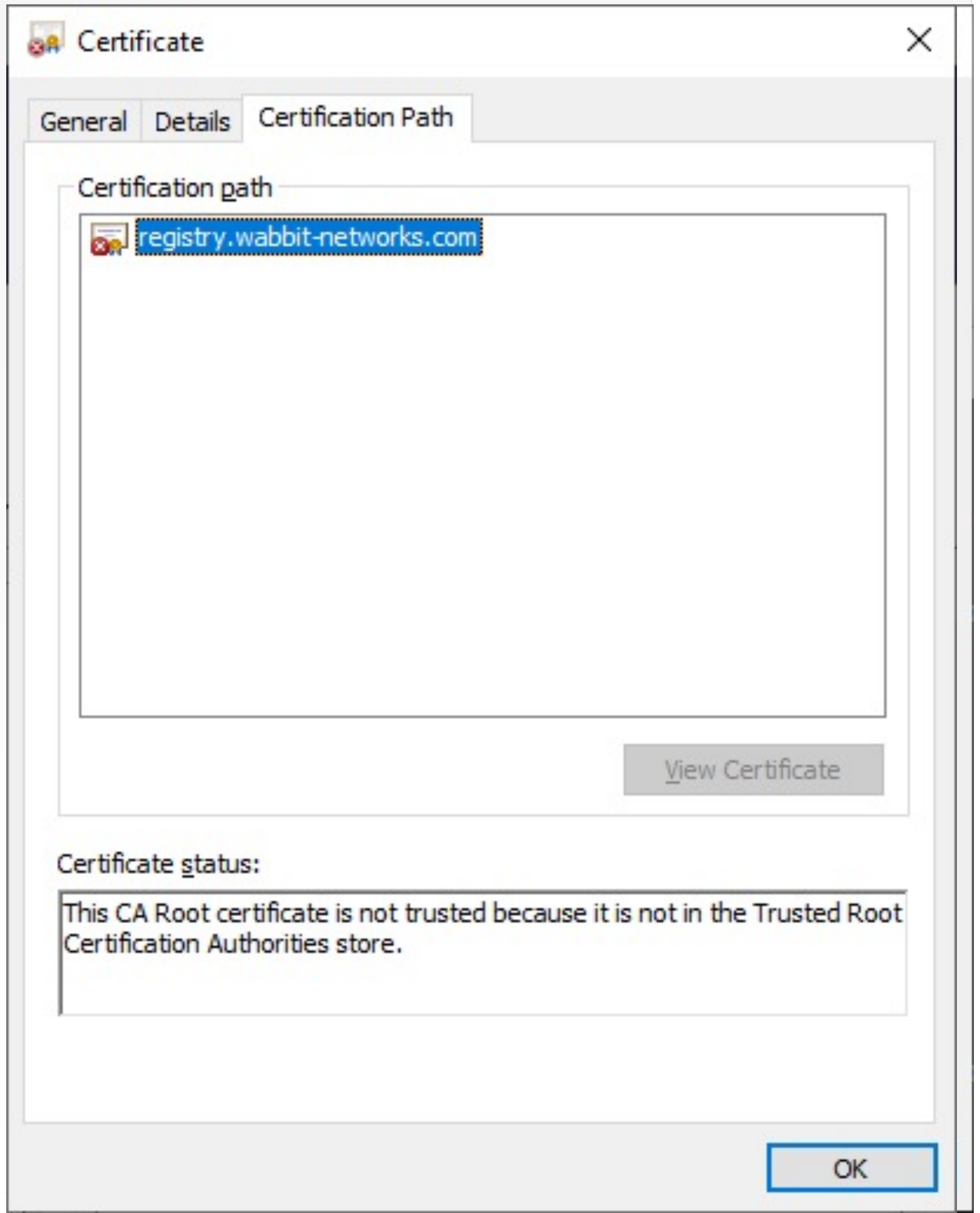


Author: Nho Luong
Skill: DevOps Engineer Lead

Key – x509

- Generate an x509 Cert
    - Subject CN = originating/vendor registry

```
openssl req \
-x509 \
-sha256 \
-nodes \
-newkey rsa:2048 \
-days 365 \
-subj "/CN=registry.wabbit-networks.com" \
-keyout wabbit-netowrks.key \
-out wabbit-netowrks.crt
```



This CA Root certificate is not trusted because it is not in the Trusted Root Certification Authorities store.

Author: Nho Luong
Skill: DevOps Engineer Lead

nv2 cli

```
docker build \
-t registry.wabbit-networks.com/net-monitor:v1 \
.
```

```
docker generatemanifest \
registry.wabbit-networks.com/net-monitor:v1 > net-monitor_v1-manifest.json
```

```
nv2 sign      --method x509 \
-k wabbit-networks.key \
-r registry.wabbit-networks.com/net-monitor:v1 \
-o net-monitor.signature.json \
file:net-monitor_v1-manifest.json
```

Author: Nho Luong
Skill: DevOps Engineer Lead

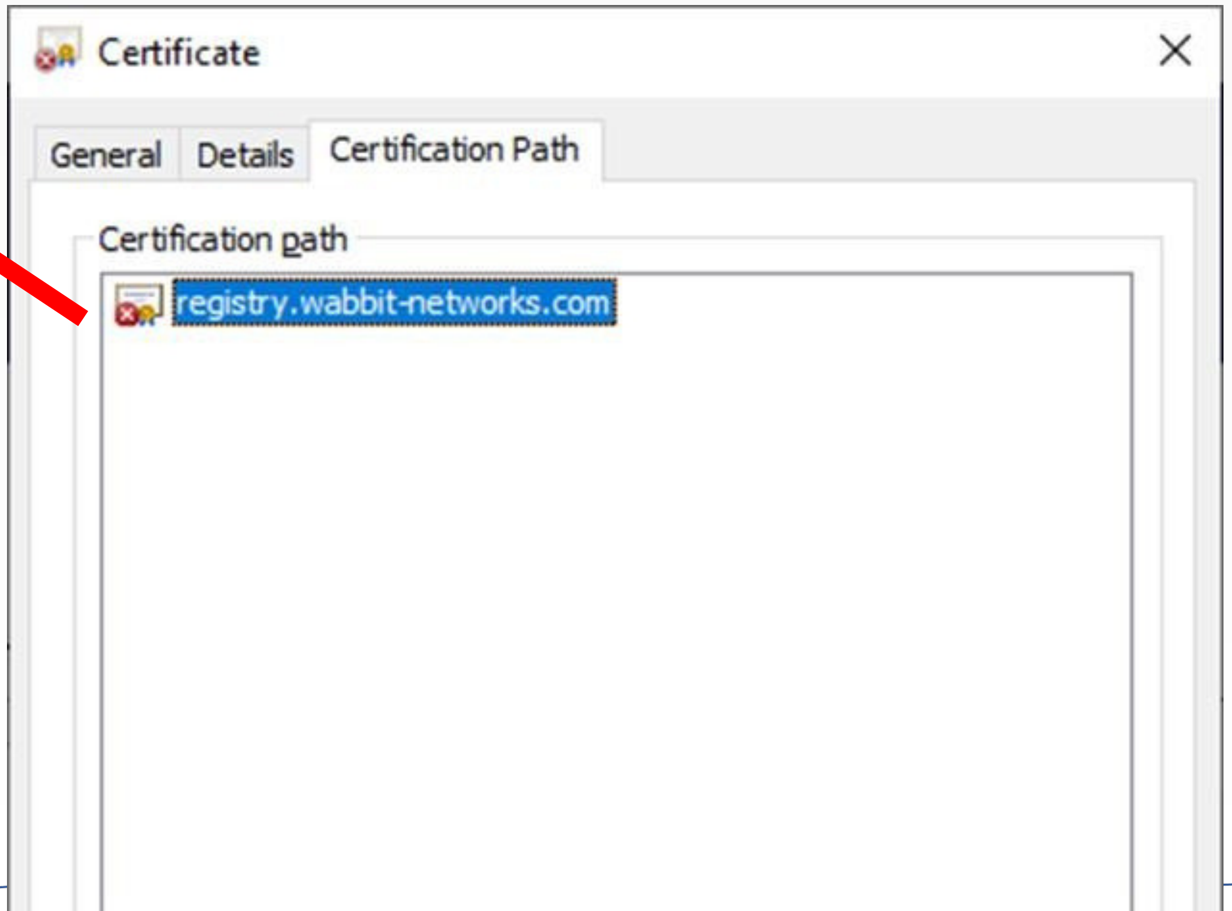
Signature

net-monitor.signature.json

Cert References

OCI Descriptor

```
{
"signed": {
"exp": 1626938793,
"nbf": 1595402793,
"iat": 1595402793,
  "mediaType": "application/vnd.oci.image.manifest.v1+json",
  "digest": "sha256:3351c53952446db17d21b86cfe5829ae70f823aff5d410fbf09dff820a39ab55",
  "size": 528,
  "references": [
  "registry.wabbit-networks.com/net-monitor:latest",  "registry.wabbit-networks.com/net-
  monitor :v1"
  ]
  },
```



Author: Nho Luong
Skill: DevOps Engineer Lead

Signature

net-monitor.signature.json

```
{
"signed": {
"exp": 1626938793,
"nbf": 1595402793,
"iat": 1595402793,
"mediaType": "application/vnd.oci.image.manifest.v1+json",
"digest": "sha256:3351c53952446db17d21b86cfe5829ae70f823aff5d410fbf09dff820a39ab55", "size":
528,
"references": [
"registry.wabbit-networks.com/net-monitor:latest", "registry.wabbit-networks.com/net-
monitor :v1"
                                                                                                    ]
                                                                                                  } ,
"signature": { "typ": "x509",
"sig": "uFKaCyQ4MtVHemfLVq5gYZyeiClS20tksXzP7hhpeqqjCNK9DiHnoDpkq91sutLqd1o6RCxpfFVuGXy20oqRu1/ZoXXAVC3y7lS6z/wqJ4VDB
KSj/H6xyYn7pH3GE8GHR6kjFPqrGsl/OS4yYH2oNXEm9W8Pju2wC381+FCgf4LNf7k6u2Uf4Fb0/Fl40qzvr0m2Fv5pXtRY+wdJctqJb+t408VcXJkNj0U7xoOe0zU
r3l1A6xLYqjd0ZY08JBQ8FQul0Vpxrmg0Xdtwd/wEolvia48lxD1x7yphW5bFvJOTd62rOJgd4uI7jYJF3ZLmwjY+geMk5e6Wkp5OyXGjXw==",
"alg": "RS256", "x5c": [
          "MIIDmzCCAoOgAw IBAgIUFSzsIT 4/pKtGzywuZWWE7ydi LBIwDQYJKoZIhvcNAQELBQAw XTELMAkGA1UEBhMCQVUxEzARBgNVBAgMClNvbWUtU3 RhdGUx ITAfBg NV BAoMGEludGVybmV0 IFdpZGdpdHMg UHR5IEx0ZDEWMB
 QGA1UEAww NKi5le GFtcGxlLmNvb TAeFw0 yMDA3MjIwMzA 2MTBaFw0yMTA 3MjIwMzA2MTBaMF0xCzAJBgNVBAYTAkFVMRMwEQYDVQQIDAp Tb21lLVN0YXRlMSEwHwYDVQQKDBhJbnRlcm5 ldCBXa WRnaXRzIFB0e SBMdGQxFjAUBgNVBAMMD
 SouZXhhbXBsZS5jb20wgg EiMA0GCSqGSIb3DQEBAQUAA4 IBDwAwggEKAo IBAQDM 0MNLy/ f1SyRM 0ZQu3AtJnCU3 O5x8nn OeV1my SmZNr2 SCqR8+ jENAoKE5FrrSi2ffMn FPP/7DqGnbb9 +b1nD9 ucFNsI 1iW7Ir F/GlqOM7jJhUMNnOyatz
8mddtQgXr 3SZ9bigbc/lxuVGacvi64DewoWzMFr 4ZMGq8 ik7aDn HryUDw XJFE+KGNbsRe O1ePqKmPiLvk LG4sBTqeTuCk+Grrr5 t1COujwuFWfh MjmRfq 34QGqUZ3SHJYXPzOAxgV3fCm BP9IgHuSv/b1 udx5Htf1BV7WlARtXf
E216…"
]
}
}
```

Author: Nho Luong
Skill: DevOps Engineer Lead

Next

- Persisted as an OCI Artifact

```
"config.mediaType": "application/vnd.cncf.notary.config.v2+json"

oras push registry.wabbit-networks.com/net-monitor:v1 \
--manifest-config net-monitor.signature.json:application/vnd.cncf.notary.config.v2+json
```

OCI Manifest

```
{
"schemaVersion": 2, "config": {
"mediaType": "application/vnd.cncf.notary.config.v2+json",
"digest": "sha256:c7848182f2c817415f0de63206f9e4220012cbb0bdb750c2ecf8020350239814", "size":
1906
},
"layers": []
}
```

Key management

- Key management working group is meeting on Fridays
- The prototype we just talked about uses x509
  - However, x509 keys are not currently widely accessible outside large organizations
  - Unlike for TLS there is less infra for keys, you can't use Letsencrypt keys for signing
  - Gives a binding between org name and signature
  - Can we get that via other means effectively?
- Some people want to use GPG
  - Outside Debian, the web of trust is mostly dead
  - Covid ends that model? Never realistically worked
- Ad hoc keys most likely, as used by TUF
  - You need to define how you choose to trust keys
  - Definitely not Notary v1 TOFU
  - This requires configuration and work from users, so we need to make this extremely easy
- Definitely want to be able to manage keys with existing tools
  - Cloud key stores, Vault, Parsec, Yubikeys

Prototype Roadmap

- Mapping TUF into OCI registry types
    - The canonical TUF design is for a set of files in a filesystem
    - The OCI registry objects have a slightly different design
        - For example an OCI descriptor includes a mime type
        - If we use external signature objects (not inline as in TUF) this changes the layout a little too
        - This is all fine so long as it is exactly equivalent to preserve security properties
    - The are several options to explore here, the main constraint is that registries tends to use OCI manifests for garbage collection control
- Once we have a representation, there are still more design decisions
    - Scope of TUF repository: registry, org or repo?
        - Notary v1 chose repo, which was a bad design
        - The TUF team believe that registry is the right scope
        - Some of the registry operators think that is too large
        - Affects key delegations and root of trust

More design work

- Ongoing discussion about rollback protection
- Ephemeral machines don't have a history of the repository state, so if an attacker deletes history they won't notice
    - Potential solution is to regularly update client base images with the repository state; the most generic solution but also requires work
    - Another solution is to use transparency logs as a public record of the state of the world; there is a difficulty though in that these are easiest to use with public data, and they are additional infrastructure that needs to be maintained outside the registry
- Ephemeral infrastructure has huge advantages, but it does impact security so we need to think about the consequences

Issues about use of registry

- The Update Framework is concerned with updates…
- We don't have a good exposure of what updates are in a registry
- We do not tend to delete much content as it is also an archival record, and we want to support rollbacks and clients that have not yet updated
- So a repository will have a lot of tags in…
    - There are currently 386 tags for Ubuntu in Docker Hub…
    - 14.04, 16.04, 18.04, 20.04 and 20.10 and what those point to are current
    - But we discourage use of latest and generic tags, and many people want immutable tags
    - This means additional information is needed to understand what an update is, eg semver, or external tooling which describes the versioning
- I think we made some design mistakes here, but rectifying will be difficult

Thank You

Author: Nho Luong
Skill: DevOps Engineer Lead