KubeCon CloudNativeCon North America 2022
BUILDING FOR THE ROAD AHEAD
DETROIT 2022

SBOM X-Ray Superpowers

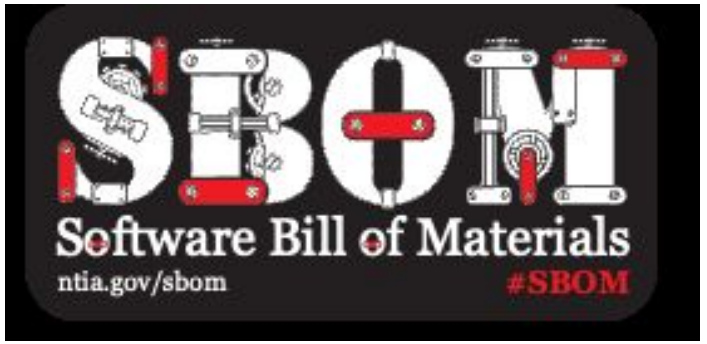
Brandon Lum (@lumjjb), Google
Christopher Phillips (@spiffcs), Anchore

Contributions by: Marco Deicas (Google)

SBOM, an ingredient list

Software Bill of Materials (SBOM):
A nested inventory for software

- EO and OMB Memo



Ingredients/components can be used to help evaluate risk appetite:

- What vulnerabilities does this have?
- Are any of these components affected by known compromises?
- Do any of the components contain non compliant licenses?

How do you get an ingredient list

peanut allergy

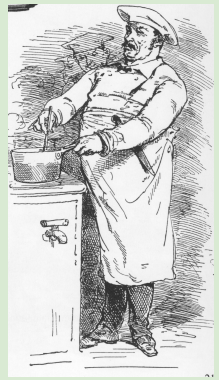
does this contain peanuts?

Option #1

Take a look at it - no visible peanuts
Get a friend to taste it, doesn't taste like there's peanuts

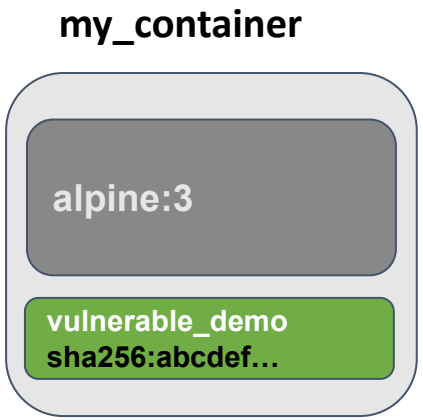
Option #2

Chef: Ah yes, we boil peanuts when making the broth, that's our special secret!



Cooking analogy

Option #1



Software Composition Analysis (SCA)
Package metadata
Reading Binary Symbols
Heuristics/Fingerprinting

Cooking analogy

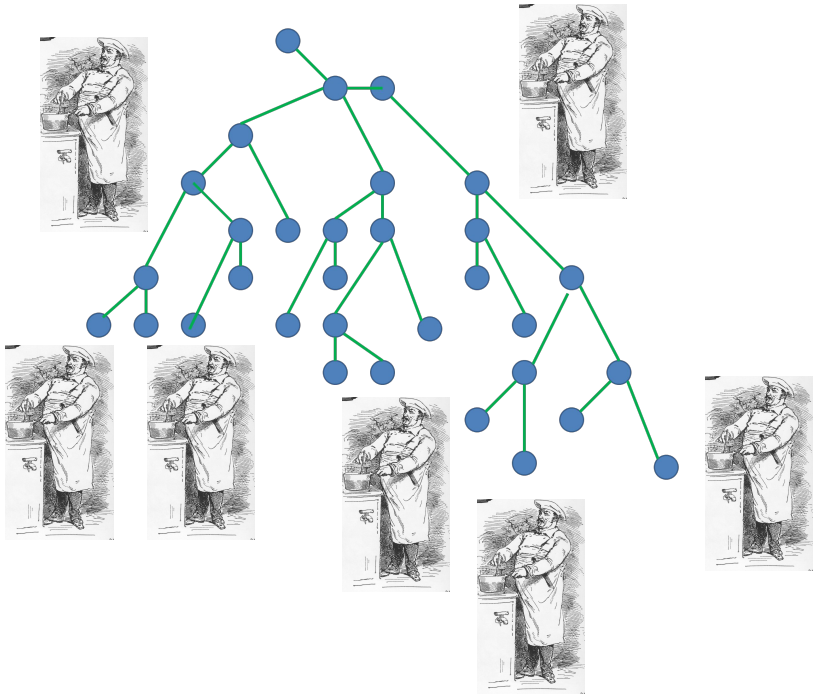
Option #1

my_container

Software Composition Analysis (SCA)
Package metadata
Reading Binary Symbols
Heuristics/Fingerprinting

Option #2

Vendor SBOMs
Build Metadata
etc.



Cooking analogy

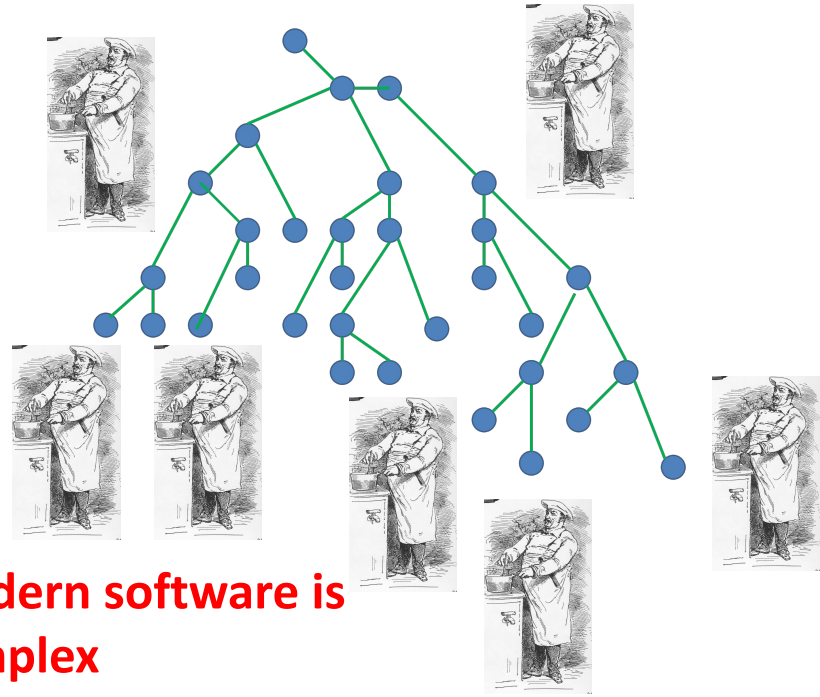
Option #1

my_container

Software Composition Analysis (SCA)
Package metadata
Reading Binary Symbols
Heuristics/Fingerprinting

Option #2

Vendor SBOMs
Build Metadata
etc.



Modern software is complex

$$$$$

Security vs Usability

In the world of SBOMs today, observation and analysis is still a leading method of SBOM generation, why?

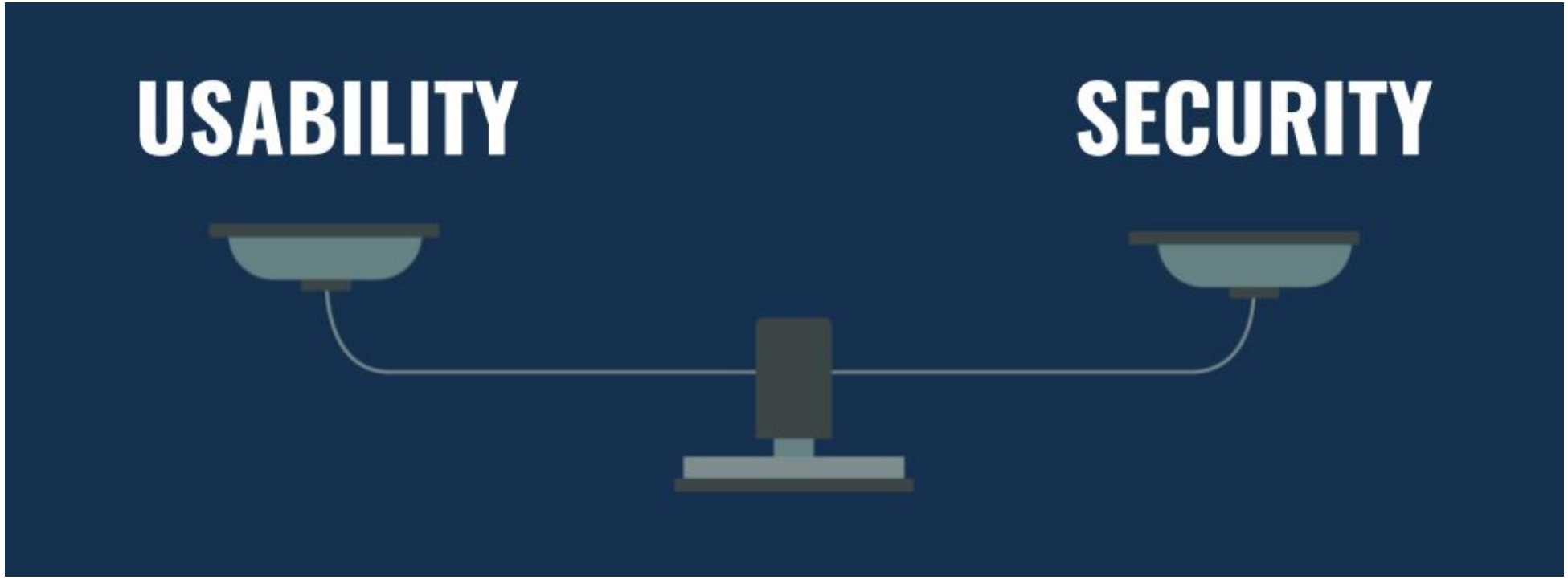
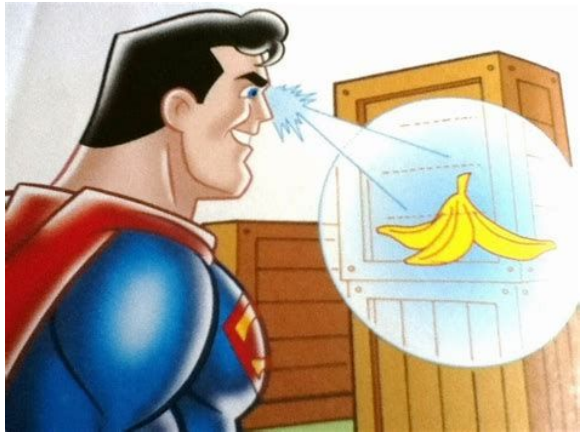


Image Credit: https://medium.com/mycrypto/the-impossible-balance-between-usability-security-55000c9fc46d

Striking a balance

Sifting through the data at different stages

- **Build Time**
  - A great chance to capture the entire system
  - Manifests, Package DB, and other metadata are all still available
  - Relationships from build tools, environment, and composition is still captures



- **Post Build**
  - Possibility of data loss not being able to scan prior dependencies
  - Compilation itself is lossy
  - Metadata after the fact is not available

Generating a Software Bill of Material

Example:

Scanning an artifact

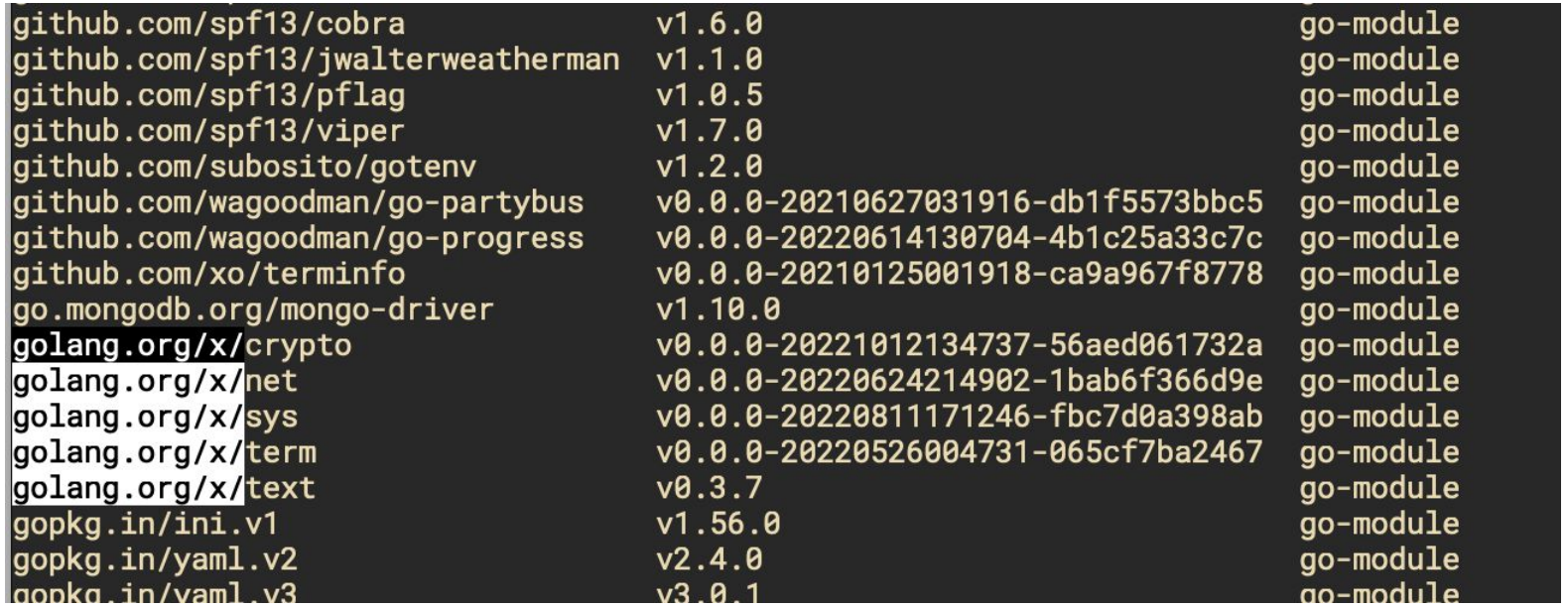
```
I ~/d/s/rke syft file:build
                                                    Tue Oct 11 15:09:49 2022
 ✔ Indexed build
 ✔ Cataloged packages      [121 packages]
NAME                                    VERSION                                  TYPE
github.com/MakeNowJust/heredoc          v0.0.0-20170808103936-bb23615498cd       go-module
github.com/Masterminds/goutils          v1.1.1                                   go-module
github.com/Masterminds/semver/v3        v3.1.1                                   go-module
github.com/Masterminds/sprig/v3         v3.2.2                                   go-module
github.com/PuerkitoBio/purell           v1.1.1                                   go-module
github.com/PuerkitoBio/urlesc           v0.0.0-20170810143723-de5bf2ad4578        go-module
github.com/apparentlymart/go-cidr       v1.0.1                                   go-module
github.com/aws/aws-sdk-go               v1.38.65                                 go-module
github.com/beorn7/perks                 v1.0.1                                   go-module
github.com/blang/semver                 v3.5.1+incompatible                      go-module
github.com/cespare/xxhash/v2            v2.1.2                                   go-module
github.com/chai2010/gettext-go          v0.0.0-20160711120539-c6fed771bfd5       go-module
github.com/containerd/containerd        v1.5.13                                  go-module
```

Scanning its source

```
I ~/d/s/r/rke release/v1.4 syft dir:.
 2022
 ✔ Indexed .
 ✔ Cataloged packages      [140 packages]
NAME                                    VERSION                                  TYPE
github.com/Azure/go-ansiterm            v0.0.0-20210617225240-d185dfc1b5a1       go-module
github.com/MakeNowJust/heredoc          v0.0.0-20170808103936-bb23615498cd       go-module
github.com/Masterminds/goutils          v1.1.1                                   go-module
github.com/Masterminds/semver/v3        v3.1.1                                   go-module
github.com/Masterminds/sprig/v3         v3.2.2                                   go-module
github.com/Microsoft/go-winio           v0.5.2                                   go-module
github.com/Microsoft/hcsshim            v0.9.3                                   go-module
github.com/PuerkitoBio/purell           v1.1.1                                   go-module
github.com/PuerkitoBio/urlesc           v0.0.0-20170810143723-de5bf2ad4578        go-module
github.com/apparentlymart/go-cidr       v1.0.1                                   go-module
github.com/aws/aws-sdk-go               v1.38.65                                 go-module
github.com/beorn7/perks                 v1.0.1                                   go-module
```

Generating a Software Bill of Material

Is that everything in the new dependency?

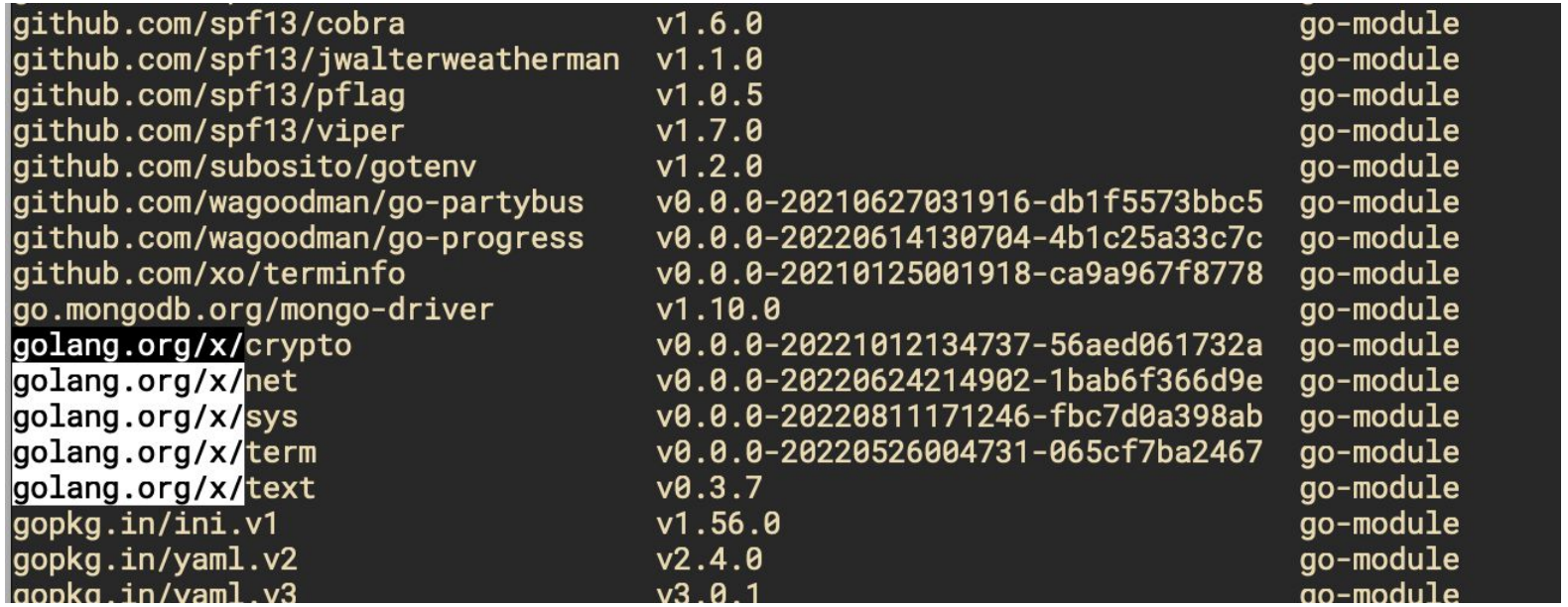


github.com/spf13/cobra	v1.6.0	go-module
github.com/spf13/jwalterweatherman	v1.1.0	go-module
github.com/spf13/pflag	v1.0.5	go-module
github.com/spf13/viper	v1.7.0	go-module
github.com/subosito/gotenv	v1.2.0	go-module
github.com/wagoodman/go-partybus	v0.0.0-20210627031916-db1f5573bbc5	go-module
github.com/wagoodman/go-progress	v0.0.0-20220614130704-4b1c25a33c7c	go-module
github.com/xo/terminfo	v0.0.0-20210125001918-ca9a967f8778	go-module
go.mongodb.org/mongo-driver	v1.10.0	go-module
golang.org/x/crypto	v0.0.0-20221012134737-56aed061732a	go-module
golang.org/x/net	v0.0.0-20220624214902-1bab6f366d9e	go-module
golang.org/x/sys	v0.0.0-20220811171246-fbc7d0a398ab	go-module
golang.org/x/term	v0.0.0-20220526004731-065cf7ba2467	go-module
golang.org/x/text	v0.3.7	go-module
gopkg.in/ini.v1	v1.56.0	go-module
gopkg.in/yaml.v2	v2.4.0	go-module
gopkg.in/yaml.v3	v3.0.1	go-module

Generating a Software Bill of Material

Is that everything in the new dependency?

I just scanned what was downloaded - seems good?



github.com/spf13/cobra	v1.6.0	go-module
github.com/spf13/jwalterweatherman	v1.1.0	go-module
github.com/spf13/pflag	v1.0.5	go-module
github.com/spf13/viper	v1.7.0	go-module
github.com/subosito/gotenv	v1.2.0	go-module
github.com/wagoodman/go-partybus	v0.0.0-20210627031916-db1f5573bbc5	go-module
github.com/wagoodman/go-progress	v0.0.0-20220614130704-4b1c25a33c7c	go-module
github.com/xo/terminfo	v0.0.0-20210125001918-ca9a967f8778	go-module
go.mongodb.org/mongo-driver	v1.10.0	go-module
golang.org/x/crypto	v0.0.0-20221012134737-56aed061732a	go-module
golang.org/x/net	v0.0.0-20220624214902-1bab6f366d9e	go-module
golang.org/x/sys	v0.0.0-20220811171246-fbc7d0a398ab	go-module
golang.org/x/term	v0.0.0-20220526004731-065cf7ba2467	go-module
golang.org/x/text	v0.3.7	go-module
gopkg.in/ini.v1	v1.56.0	go-module
gopkg.in/yaml.v2	v2.4.0	go-module
gopkg.in/yaml.v3	v3.0.1	go-module

Generating a Software Bill of Material

Is that everything in the new dependency?

I just scanned what was downloaded - seems good?

LGTM!

```
github.com/spf13/cobra                  v1.6.0                                  go-module
github.com/spf13/jwalterweatherman      v1.1.0                                  go-module
github.com/spf13/pflag                  v1.0.5                                  go-module
github.com/spf13/viper                  v1.7.0                                  go-module
github.com/subosito/gotenv              v1.2.0                                  go-module
github.com/wagoodman/go-partybus        v0.0.0-20210627031916-db1f5573bbc5      go-module
github.com/wagoodman/go-progress        v0.0.0-20220614130704-4b1c25a33c7c      go-module
github.com/xo/terminfo                  v0.0.0-20210125001918-ca9a967f8778      go-module
go.mongodb.org/mongo-driver             v1.10.0                                 go-module
golang.org/x/crypto                     v0.0.0-20221012134737-56aed061732a      go-module
golang.org/x/net                        v0.0.0-20220624214902-1bab6f366d9e      go-module
golang.org/x/sys                        v0.0.0-20220811171246-fbc7d0a398ab      go-module
golang.org/x/term                       v0.0.0-20220526004731-065cf7ba2467      go-module
golang.org/x/text                       v0.3.7                                  go-module
gopkg.in/ini.v1                         v1.56.0                                 go-module
gopkg.in/yaml.v2                        v2.4.0                                  go-module
gopkg.in/yaml.v3                        v3.0.1                                  go-module
```

OOPS

But wait —

OOPS

But wait — There’s More

OOPS

But wait — There's More


```
    "version": "v0.0.0-20211110154304-99a53858aa08",
    "type": "go-module",
    "foundBy": "go-module-binary-cataloger",
    "locations": [
     {
      "path": ".git/lfs/objects/ad/5f/ad5ff64cef84393936e1222db0ac10bc840766d83573cf747082a4a36662e46c"
     }
    ],
    "licenses": [],
    "language": "go",
    "cpes": [
     "cpe:2.3:a:golang:x\\/sys:v0.0.0-20211110154304-99a53858aa08:*:*:*:*:*:*:*"
    ],
    "purl": "pkg:golang/golang.org/x/sys@v0.0.0-20211110154304-99a53858aa08",
    "metadataType": "GolangBinMetadata",
    "metadata": {
     "goCompiledVersion": "go1.17.1",
     "architecture": "amd64",
     "h1Digest": "h1:WecRHqgE09JBkh/584XIE6PMz5KKE/vER4izNUi30AQ=",
     "mainModule": "github.com/anchore/syft"
    }
   },
   {
    "id": "67251088704e6ad4",
    "name": "golang.org/x/sys",
    "version": "v0.0.0-20211216021012-1d35b9e2eb4e",
    "type": "go-module",
    "foundBy": "go-module-binary-cataloger",
    "locations": [
     {
      "path": ".tmp/chronicle"
     }
    ],
    "licenses": [],
    "language": "go",
    "cpes": [
     "cpe:2.3:a:golang:x\\/sys:v0.0.0-20211216021012-1d35b9e2eb4e:*:*:*:*:*:*:*"
    ],
    "purl": "pkg:golang/golang.org/x/sys@v0.0.0-20211216021012-1d35b9e2eb4e",
    "metadataType": "GolangBinMetadata",
    "metadata": {
     "goCompiledVersion": "go1.17.13",
     "architecture": "amd64",
     "h1Digest": "h1:fLOSk5Q00efkSvAm+4xcoXD+RRmLmmulPn5I3Y9F2EM=",
     "mainModule": "github.com/anchore/chronicle"
    }
   },
   {
    "id": "d3bdaccdf06d9412",
    "name": "golang.org/x/sys",
    "version": "v0.0.0-20211216021012-1d35b9e2eb4e",
    "type": "go-module",
    "foundBy": "go-module-binary-cataloger",
    "locations": [
     {
      "path": ".tmp/chronicle"
     }
    ]
```

OOPS

But wait — There's More

99a538…


```
  "version": "v0.0.0-20211110154304-99a53858aa08",
  "type": "go-module",
  "foundBy": "go-module-binary-cataloger",
  "locations": [
   {
    "path": ".git/lfs/objects/ad/5f/ad5ff64cef84393936e1222db0ac10bc840766d83573cf747082a4a36662e46c"
   }
  ],
  "licenses": [],
  "language": "go",
  "cpes": [
   "cpe:2.3:a:golang:x\\/sys:v0.0.0-20211110154304-99a53858aa08:*:*:*:*:*:*:*"
  ],
  "purl": "pkg:golang/golang.org/x/sys@v0.0.0-20211110154304-99a53858aa08",
  "metadataType": "GolangBinMetadata",
  "metadata": {
   "goCompiledVersion": "go1.17.1",
   "architecture": "amd64",
   "h1Digest": "h1:WecRHqgE09JBkh/584XIE6PMz5KKE/vER4izNUi30AQ=",
   "mainModule": "github.com/anchore/syft"
  }
 },
 {
  "id": "67251088704e6ad4",
  "name": "golang.org/x/sys",
  "version": "v0.0.0-20211216021012-1d35b9e2eb4e",
  "type": "go-module",
  "foundBy": "go-module-binary-cataloger",
  "locations": [
   {
    "path": ".tmp/chronicle"
   }
  ],
  "licenses": [],
  "language": "go",
  "cpes": [
   "cpe:2.3:a:golang:x\\/sys:v0.0.0-20211216021012-1d35b9e2eb4e:*:*:*:*:*:*:*"
  ],
  "purl": "pkg:golang/golang.org/x/sys@v0.0.0-20211216021012-1d35b9e2eb4e",
  "metadataType": "GolangBinMetadata",
  "metadata": {
   "goCompiledVersion": "go1.17.13",
   "architecture": "amd64",
   "h1Digest": "h1:fLOSk5Q00efkSvAm+4xcoXD+RRmLmmulPn5I3Y9F2EM=",
   "mainModule": "github.com/anchore/chronicle"
  }
 },
 {
  "id": "d3bdaccdf06d9412",
  "name": "golang.org/x/sys",
  "version": "v0.0.0-20211216021012-1d35b9e2eb4e",
  "type": "go-module",
  "foundBy": "go-module-binary-cataloger",
  "locations": [
   {
    "path": ".tmp/chronicle"
   }
```

OOPS

But wait — There's More

99a538…

1d35b9e…

```
  "version": "v0.0.0-20211110154304-99a53858aa08",
  "type": "go-module",
  "foundBy": "go-module-binary-cataloger",
  "locations": [
   {
    "path": ".git/lfs/objects/ad/5f/ad5ff64cef84393936e1222db0ac10bc840766d83573cf747082a4a36662e46c"
   }
  ],
  "licenses": [],
  "language": "go",
  "cpes": [
   "cpe:2.3:a:golang:x\\/sys:v0.0.0-20211110154304-99a53858aa08:*:*:*:*:*:*:*"
  ],
  "purl": "pkg:golang/golang.org/x/sys@v0.0.0-20211110154304-99a53858aa08",
  "metadataType": "GolangBinMetadata",
  "metadata": {
   "goCompiledVersion": "go1.17.1",
   "architecture": "amd64",
   "h1Digest": "h1:WecRHqgE09JBkh/584XIE6PMz5KKE/vER4izNUi30AQ=",
   "mainModule": "github.com/anchore/syft"
  }
 },
 {
  "id": "67251088704e6ad4",
  "name": "golang.org/x/sys",
  "version": "v0.0.0-20211216021012-1d35b9e2eb4e",
  "type": "go-module",
  "foundBy": "go-module-binary-cataloger",
  "locations": [
   {
    "path": ".tmp/chronicle"
   }
  ],
  "licenses": [],
  "language": "go",
  "cpes": [
   "cpe:2.3:a:golang:x\\/sys:v0.0.0-20211216021012-1d35b9e2eb4e:*:*:*:*:*:*:*"
  ],
  "purl": "pkg:golang/golang.org/x/sys@v0.0.0-20211216021012-1d35b9e2eb4e",
  "metadataType": "GolangBinMetadata",
  "metadata": {
   "goCompiledVersion": "go1.17.13",
   "architecture": "amd64",
   "h1Digest": "h1:fLOSk5Q00efkSvAm+4xcoXD+RRmLmmulPn5I3Y9F2EM=",
   "mainModule": "github.com/anchore/chronicle"
  }
 },
 {
  "id": "d3bdaccdf06d9412",
  "name": "golang.org/x/sys",
  "version": "v0.0.0-20211216021012-1d35b9e2eb4e",
  "type": "go-module",
  "foundBy": "go-module-binary-cataloger",
  "locations": [
   {
    "path": ".tmp/chronicle"
   }
  ]
```

OOPS

But wait — There's More

99a538…

1d35b9e…

fbc7d0a…

And More


```
  "version": "v0.0.0-20211110154304-99a53858aa08",
  "type": "go-module",
  "foundBy": "go-module-binary-cataloger",
  "locations": [
   {
    "path": ".git/lfs/objects/ad/5f/ad5ff64cef84393936e1222db0ac10bc840766d83573cf747082a4a36662e46c"
   }
  ],
  "licenses": [],
  "language": "go",
  "cpes": [
   "cpe:2.3:a:golang:x\\/sys:v0.0.0-20211110154304-99a53858aa08:*:*:*:*:*:*:*"
  ],
  "purl": "pkg:golang/golang.org/x/sys@v0.0.0-20211110154304-99a53858aa08",
  "metadataType": "GolangBinMetadata",
  "metadata": {
   "goCompiledVersion": "go1.17.1",
   "architecture": "amd64",
   "h1Digest": "h1:WecRHqgE09JBkh/584XIE6PMz5KKE/vER4izNUi30AQ=",
   "mainModule": "github.com/anchore/syft"
  }
 },
 {
  "id": "67251088704e6ad4",
  "name": "golang.org/x/sys",
  "version": "v0.0.0-20211216021012-1d35b9e2eb4e",
  "type": "go-module",
  "foundBy": "go-module-binary-cataloger",
  "locations": [
   {
    "path": ".tmp/chronicle"
   }
  ],
  "licenses": [],
  "language": "go",
  "cpes": [
   "cpe:2.3:a:golang:x\\/sys:v0.0.0-20211216021012-1d35b9e2eb4e:*:*:*:*:*:*:*"
  ],
```


It’s in production isn’t it?

SBOM X-Ray Superpowers

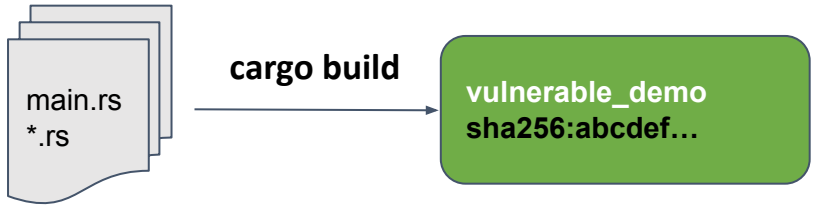
If software producers can
make their **metadata discoverable**

we can obtain more accurate and complete SBOMs
without sacrificing usability for the end user.

A Simple Containerized App

App Developer 0 builds a binary

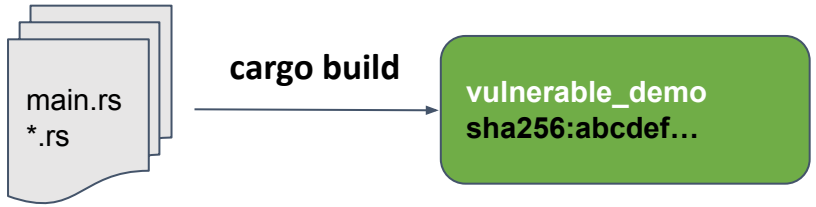
$ cargo build –release …



A Simple Containerized App

App Developer 0 builds a binary

$ cargo build –release …



App Developer 1 builds a container

$ docker build .

```
FROM alpine:3
RUN curl -sL https://github.com/… > vulnerable_demo
```

Build time vs Scan time SBOMs

Build Time

$ cargo build –release …

main.rs
*.rs

cargo build

vulnerable_demo
sha256:abcdef…

Scan Time

my_container

alpine:3

vulnerable_demo
sha256:abcdef…

syft

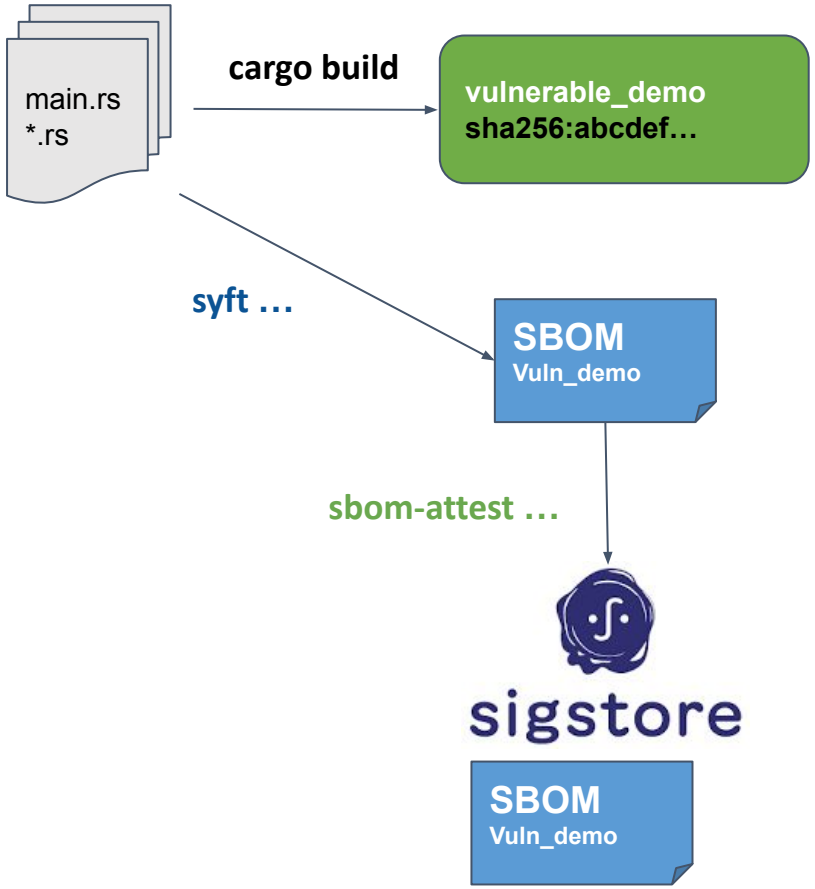
Here's your SBOM, it contains:
- alpine OS
- XYZ alpine packages
- **This "vulnerable_demo" binary (but I don't know what it is!)**

If only I could ask the "chef"...

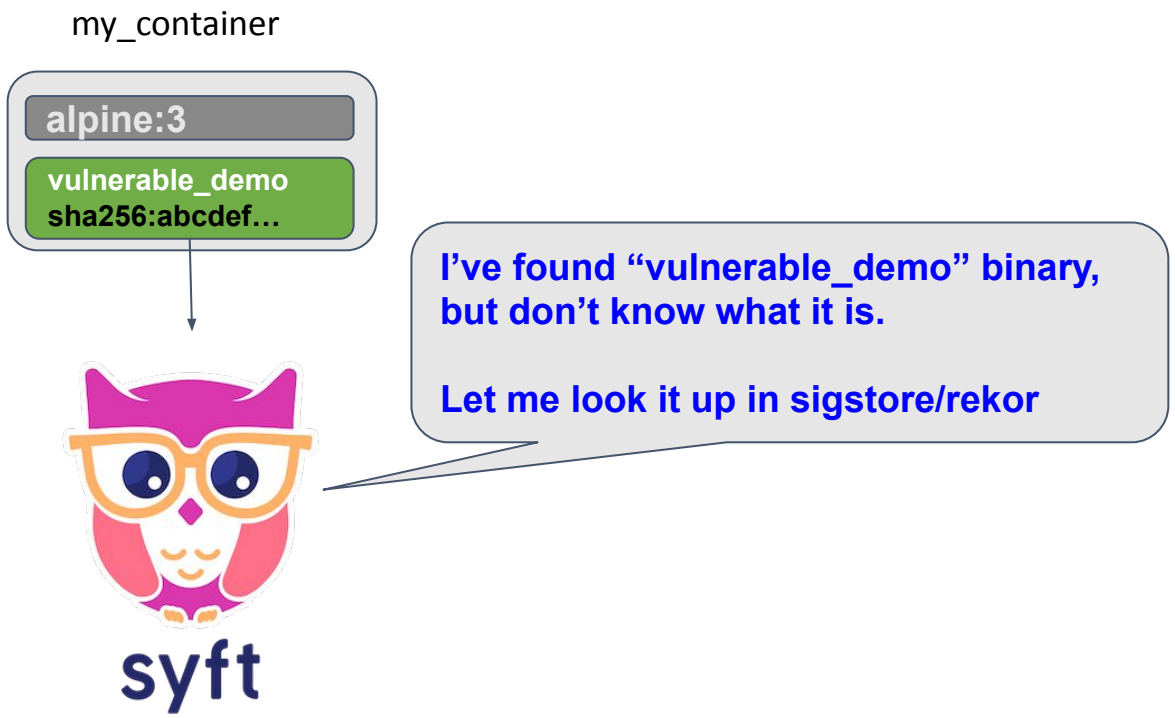
Build time vs Scan time SBOMs

Build Time

$ cargo build –release …
$ syft packages dir:. -o spdx-json > spdx.json
$ sbom-attest –sbom spdx.json …



Scan Time



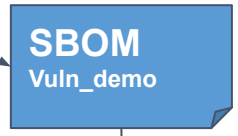
Build time vs Scan time SBOMs

Build Time

$ cargo build –release …
$ syft packages dir:. -o spdx-json > spdx.json
$ sbom-attest –sbom spdx.json …

cargo build

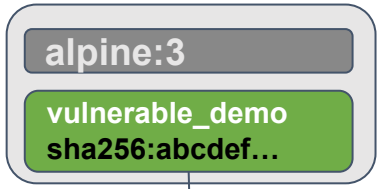
syft …



sbom-attest …

Scan Time

my_container



Any attestations for **sha256:abcdef**?

I've found "vulnerable_demo" binary, but don't know what it is.

Let me look it up in sigstore/rekor

Yup, we got an SBOM!

Build time vs Scan time SBOMs

Build Time

$ cargo build –release …
$ syft packages dir:. -o spdx-json > spdx.json
$ sbom-attest –sbom spdx.json …

main.rs
*.rs

cargo build

vulnerable_demo
sha256:abcdef…

syft …

SBOM
Vuln_demo

sbom-attest …

sigstore

SBOM
Vuln_demo

Scan Time

my_container

alpine:3

vulnerable_demo
sha256:abcdef…

Any attestations for **sha256:abcdef**?

syft

I've found "vulnerable_demo" binary, but don't know what it is.

Let me look it up in sigstore/rekor

Yup, we got an SBOM!

SBOM
Vuln_demo

Here's your SBOM, it contains:

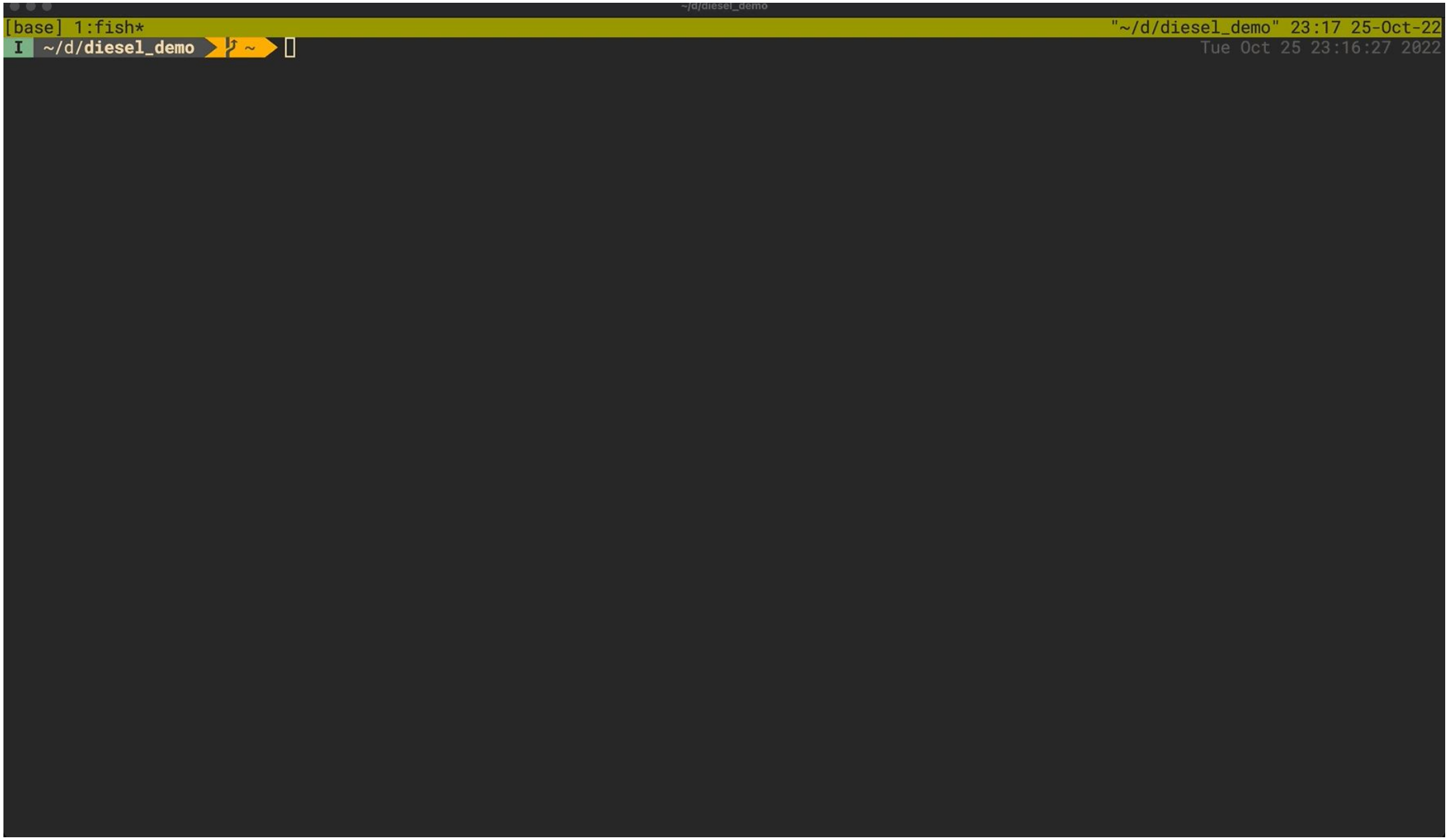
- alpine OS
- XYZ alpine packages
- **This "vulnerable_demo" binary, with corresponding SBOM**

SBOM
Vuln_demo

Demo

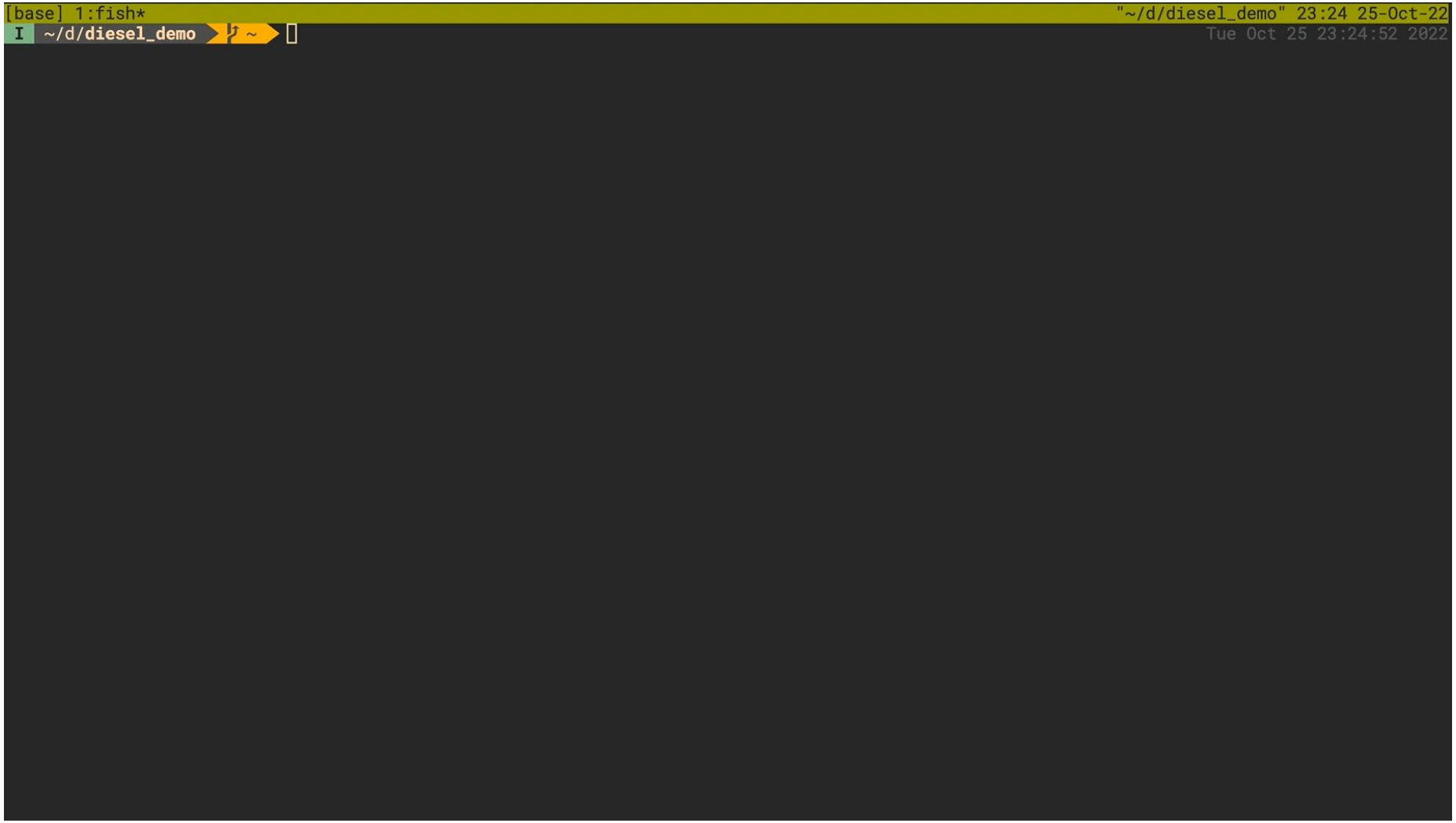
DEMO

Build the Artifact



Generate an SBOM - Vulns please!

KubeCon CloudNativeCon North America 2022



Generate an SBOM - Vulns please!


```
[base] 1:fish*                                                    "~/d/diesel_demo" 23:24 25-Oct-22
I ~/d/diesel_demo  ~                                                  Tue Oct 25 23:24:52 2022
```

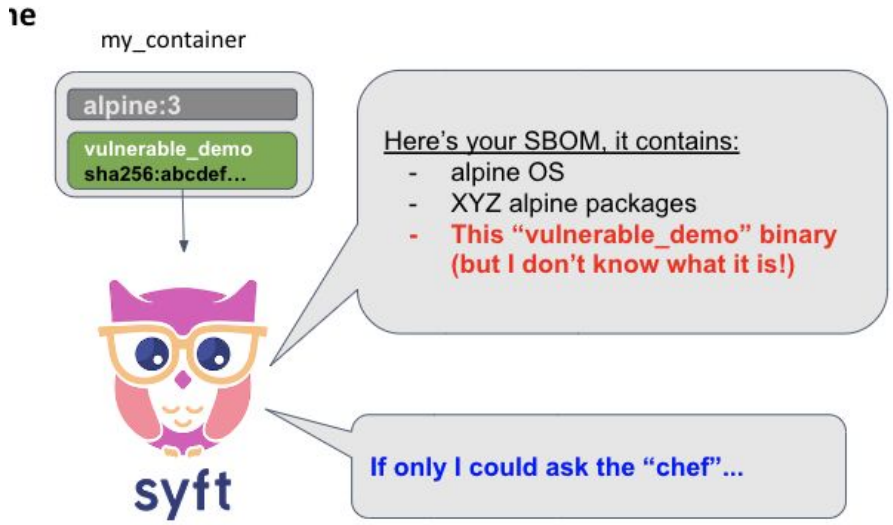
We’re good right? It says no Vulns!

Generate an Image - Vulns please!

Generate an Image - Vulns please!

KubeCon CloudNativeCon North America 2022

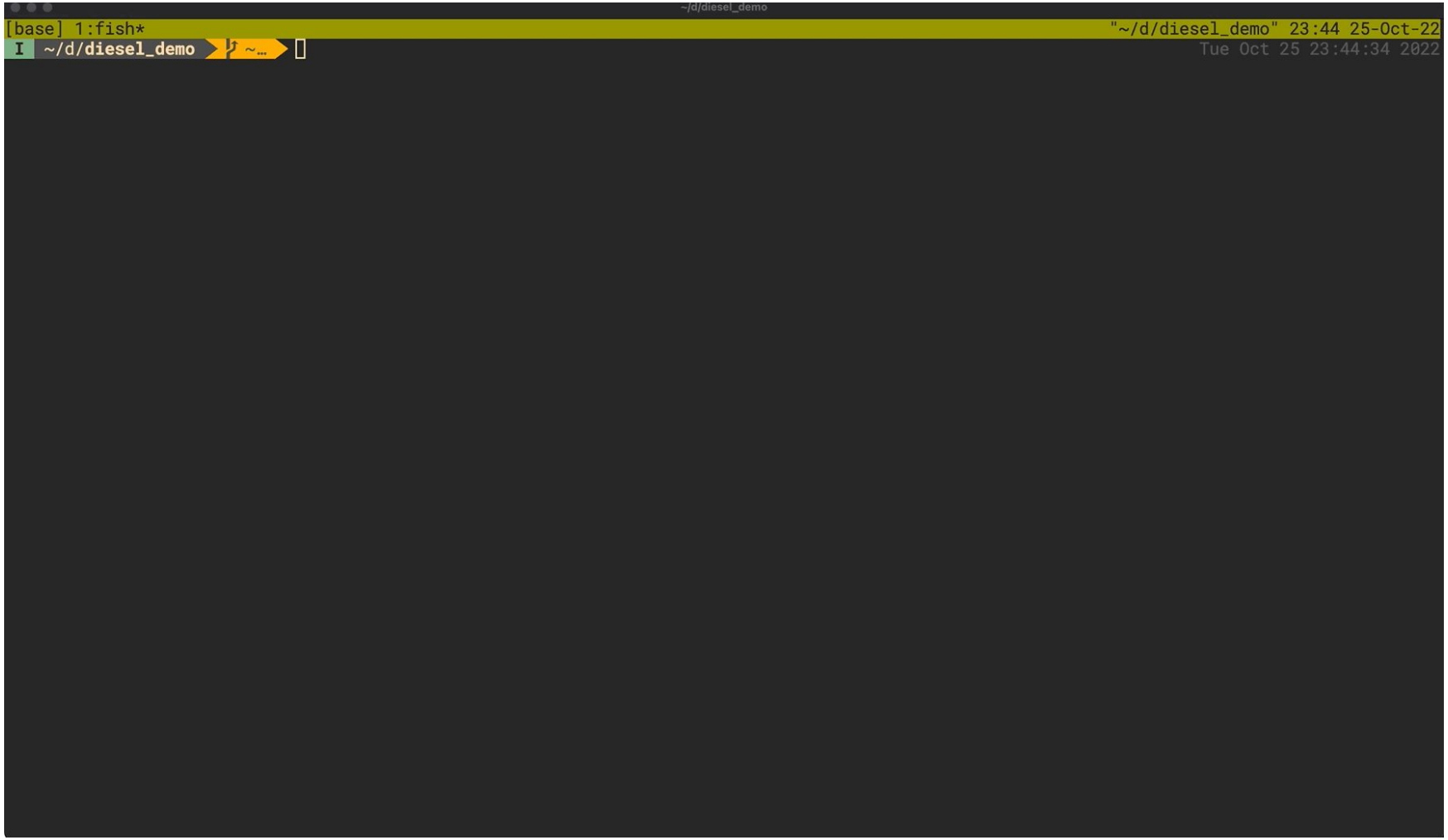
```
[base] 1:nvim  2:fish  3:fish  4:fish* 5:[tmux]-                    "~/d/diesel_demo" 08:49 26-Oct-22
I ~/d/diesel_demo  *~
                                                                    Wed Oct 26 08:49:08 2022
```




```
[base] 1:nvim  2:fish  3:fish  4:syft* 5:[tmux]-                    "~/d/diesel_demo" 08:52 26-Oct-22
I ~/d/diesel_demo  *~ docker build -t diesel_demo:1.0.0 .
                                                                    Wed Oct 26 08:52:12 2022
[+] Building 1.7s (7/7) FINISHED
 => [internal] load build definition from Dockerfile                                        0.0s
 => => transferring dockerfile: 36B                                                         0.0s
 => [internal] load .dockerignore                                                           0.0s
 => => transferring context: 2B                                                             0.0s
 => [internal] load metadata for docker.io/library/alpine:latest                            0.9s
 => [internal] load build context                                                           0.7s
 => => transferring context: 681.20kB                                                       0.6s
 => [1/2] FROM docker.io/library/alpine:latest@sha256:bc41182d7ef5ffc53a40b044e725193bc10142a1243f395ee852a8d9730fc2ad  0.0s
 => CACHED [2/2] COPY . ./target/release/show_posts                                         0.0s
 => exporting to image                                                                      0.0s
 => => exporting layers                                                                     0.0s
 => => writing image sha256:02400528729cf01a7b91bf849906eed82da375ea13dfec9ea3c605df3cae5e3f  0.0s
 => => naming to docker.io/library/diesel_demo:1.0.0                                        0.0s
I ~/d/diesel_demo  *~ syft -o json diesel_demo:1.0.0 | grype                    2373ms  Wed Oct 26 08:52:16 2022
✔ Loaded image
⠙ Parsing image
```

Bizzaro Solve Supply Chain Security

Let’s Try this again - More Fidelity



Get the sha256 of our binary

Get the sha256 of our binary

Point consumers to our SBOM


```
[base] 1:fish  2:nvim- 3:fish  4:fish*                                                    "~/d/diesel_demo" 00:42 26-Oct-22
-rw-r--r--  1 hal  staff   309K Oct 25 23:43 diesel-spdx-v0.2.0.json
-rw-r--r--  1 hal  staff   187B Oct  7 11:52 diesel.toml
drwxr-xr-x  5 hal  staff   160B Oct  7 11:52 migrations/
-rwxr-xr-x  1 hal  staff    57M Oct 26 00:28 sbom-attest*
-rw-r--r--  1 hal  staff   309K Oct 25 18:45 spdx.json
drwxr-xr-x  6 hal  staff   192B Oct 25 22:35 src/
drwxr-xr-x@ 6 hal  staff   192B Oct 25 23:24 target/
 I  ~/d/diesel_demo  ~…  shasum -a 256 ./diesel-spdx-v0.2.0.json                                  Wed Oct 26 00:38:43 202
2
86714c3b0365092c8f90fdb620292cb256f7d83dcbeeed1fd25687275fc52cd1  ./diesel-spdx-v0.2.0.json
 I  ~/d/diesel_demo  ~…  shasum -a 256 ./target/release/show_posts                    485ms  Wed Oct 26 00:38:46 202
2
dc80b5b7b4a54f6306fe88eb625595f14d70dab7bd645c4ffbdb7f650a7acab2  ./target/release/show_posts
 I  ~/d/diesel_demo  ~…  ./sbom-attest attest \                                                 Wed Oct 26 00:38:50 202
 I  ~/d/diesel_demo  ~…  ./sbom-attest attest \                                                Wed Oct 26 00:42:49 2022
                             --local \
                             --predicateType "google.com/sbom" \
                             --subjects "dc80b5b7b4a54f6306fe88eb625595f14d70dab7bd645c4ffbdb7f650a7acab2 diesel_demo" \
                             --sbomUri https://github.com/spiffcs/vulnerable_demo/releases/download/v0.4.0/diesel-spdx-v0.2.0.json \
                             --sbomSha256 86714c3b0365092c8f90fdb620292cb256f7d83dcbeeed1fd25687275fc52cd1
```

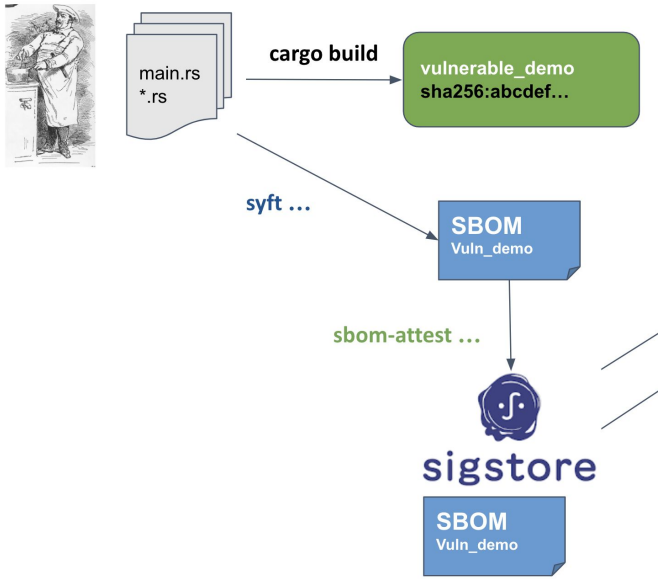
Point consumers to our SBOM

KubeCon CloudNativeCon North America 2022

```
[base] 1:fish  2:nvim- 3:fish  4:fish*                                  "~/d/diesel_demo" 00:42 26-Oct-22
-rw-r--r--  1 hal  staff   309K Oct 25 23:43 diesel-spdx-v0.2.0.json
-rw-r--r--  1 hal  staff   187B Oct  7 11:52 diesel.toml
drwxr-xr-x  5 hal  staff   160B Oct  7 11:52 migrations/
-rwxr-xr-x  1 hal  staff    57M Oct 26 00:28 sbom-attest*
-rw-r--r--  1 hal  staff   309K Oct 25 18:45 spdx.json
drwxr-xr-x  6 hal  staff   192B Oct 25 22:35 src/
drwxr-xr-x@ 6 hal  staff   192B Oct 25 23:24 target/
 I  ~/d/diesel_demo  ~…  shasum -a 256 ./diesel-spdx-v0.2.0.json                    Wed Oct 26 00:38:43 202
2
86714c3b0365092c8f90fdb620292cb256f7d83dcbeeed1fd25687275fc52cd1  ./diesel-spdx-v0.2.0.json
 I  ~/d/diesel_demo  ~…  shasum -a 256 ./target/release/show_posts          485ms  Wed Oct 26 00:38:46 202
2
dc80b5b7b4a54f6306fe88eb625595f14d70dab7bd645c4ffbdb7f650a7acab2  ./target/release/show_posts
 I  ~/d/diesel_demo  ~…  ./sbom-attest attest \                                    Wed Oct 26 00:38:50 202
 I  ~/d/diesel_demo  ~…  ./sbom-attest attest \                                    Wed Oct 26 00:42:49 2022
                         --local \
                         --predicateType "google.com/sbom" \
                         --subjects "dc80b5b7b4a54f6306fe88eb625595f14d70dab7bd645c4ffbdb7f650a7acab2 diesel_demo" \
                         --sbomUri https://github.com/spiffcs/vulnerable_demo/releases/download/v0.4.0/diesel-spdx-v0.2.0.json \
                         --sbomSha256 86714c3b0365092c8f90fdb620292cb256f7d83dcbeeed1fd25687275fc52cd1
```

Build Time

$ cargo build –release …
$ syft packages dir:. -o spdx-json > spdx.json
$ sbom-attest –sbom spdx.json …



YOU ARE HERE ^^^^^^^^

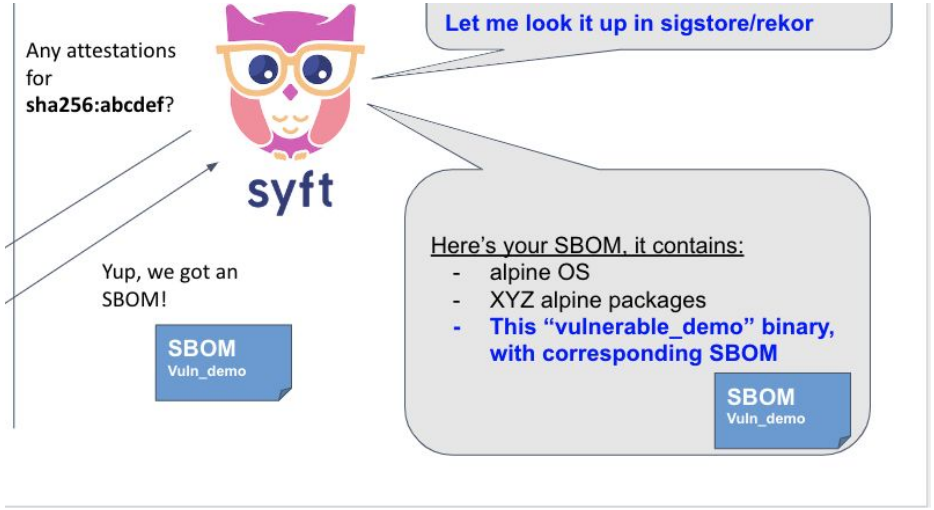
Did it work?

```
base] 1:nvim  2:nvim  3:fish  4:fish  5:fish- 6:fish*
I  ~/d/diesel_demo  ~…  syft packages -v --catalogers "all" --external-sources-enabled -o spdx-json diesel_demo:1.0.0 > spdx.json
0000]  INFO syft version: 0.59.0-SNAPSHOT-e95bd5b
0005]  INFO identified distro: Alpine Linux v3.16
0012]  INFO
                    [Rekor-cataloger]

                    This SBOM contains a relationship that references an external document. This
                    document is not present in the cataloged image or directory; rather it has
                    been found by searching the Rekor transparency log (https://www.sigstore.dev/).

                    Trusting this external document relationship requires trusting several entities:
                            - the user or CI/CD action that uploaded an entry to Rekor
                            - Rekor transparency log
                            - Fulcio CA

                    The Rekor entry(s) that were used to create the external document relationship(s)
                    are listed below by UUID. See https://github.com/sigstore/rekor for
                    information on how to query Rekor.
                            [24296fb24b8ad77ade92b102d2074f194f1efa846ff7f8a31416a9cc7afc4bbbc229c3eb71bf5302]
```



We now have a more complete SBOM!

KubeCon CloudNativeCon North America 2022

```
I ~/d/diesel_demo  *~ wget -q -O - ( \
                      rekor-cli get \
                      --uuid 24296fb24b8ad77ad9e011a61022dccf5d7fda6f827589d25b85eabebb721bf2f1004085b3b2402f | \
                      grep 'Attestation' | \
                      cut -f2 -w | \
                      jq -r .predicate.sboms[0].uri \
                      ) | grype
NAME              INSTALLED  FIXED-IN  TYPE        VULNERABILITY        SEVERITY
crossbeam-channel 0.3.9      0.4.4     rust-crate  GHSA-v5m7-53cv-f3hx  High
crossbeam-channel 0.3.9      0.4.3     rust-crate  GHSA-9g55-pg62-m8hh  High
crossbeam-utils   0.6.6      0.8.7     rust-crate  GHSA-qc84-gqf4-9926  High
crossbeam-utils   0.7.2      0.8.7     rust-crate  GHSA-qc84-gqf4-9926  High
failure           0.1.8                rust-crate  GHSA-jq66-xh47-j9f3  Critical
hyper             0.10.16              rust-crate  CVE-2021-32714       Critical
hyper             0.10.16              rust-crate  CVE-2020-35863       Critical
hyper             0.10.16              rust-crate  CVE-2021-32715       Medium
hyper             0.10.16    0.14.10   rust-crate  GHSA-f3pg-qwvg-p99c  Low
hyper             0.10.16    0.14.12   rust-crate  GHSA-f67m-9j94-qv9j  High
hyper             0.10.16    0.14.10   rust-crate  GHSA-5h46-h7hh-c6x9  Medium
metrics-util      0.5.0      0.7.0     rust-crate  GHSA-cwvc-87xq-pc5m  High
metrics-util      0.5.0      0.7.0     rust-crate  GHSA-3hxh-7jxm-59x4  Medium
owning_ref        0.4.1                rust-crate  GHSA-9qxh-258v-666c  Medium
tokio             0.1.22     1.8.4     rust-crate  GHSA-fg7r-2g4j-5cgr  High
traitobject       0.1.0                rust-crate  GHSA-pp8r-vv2j-9j5v  Critical
I ~/d/diesel_demo  *~
```

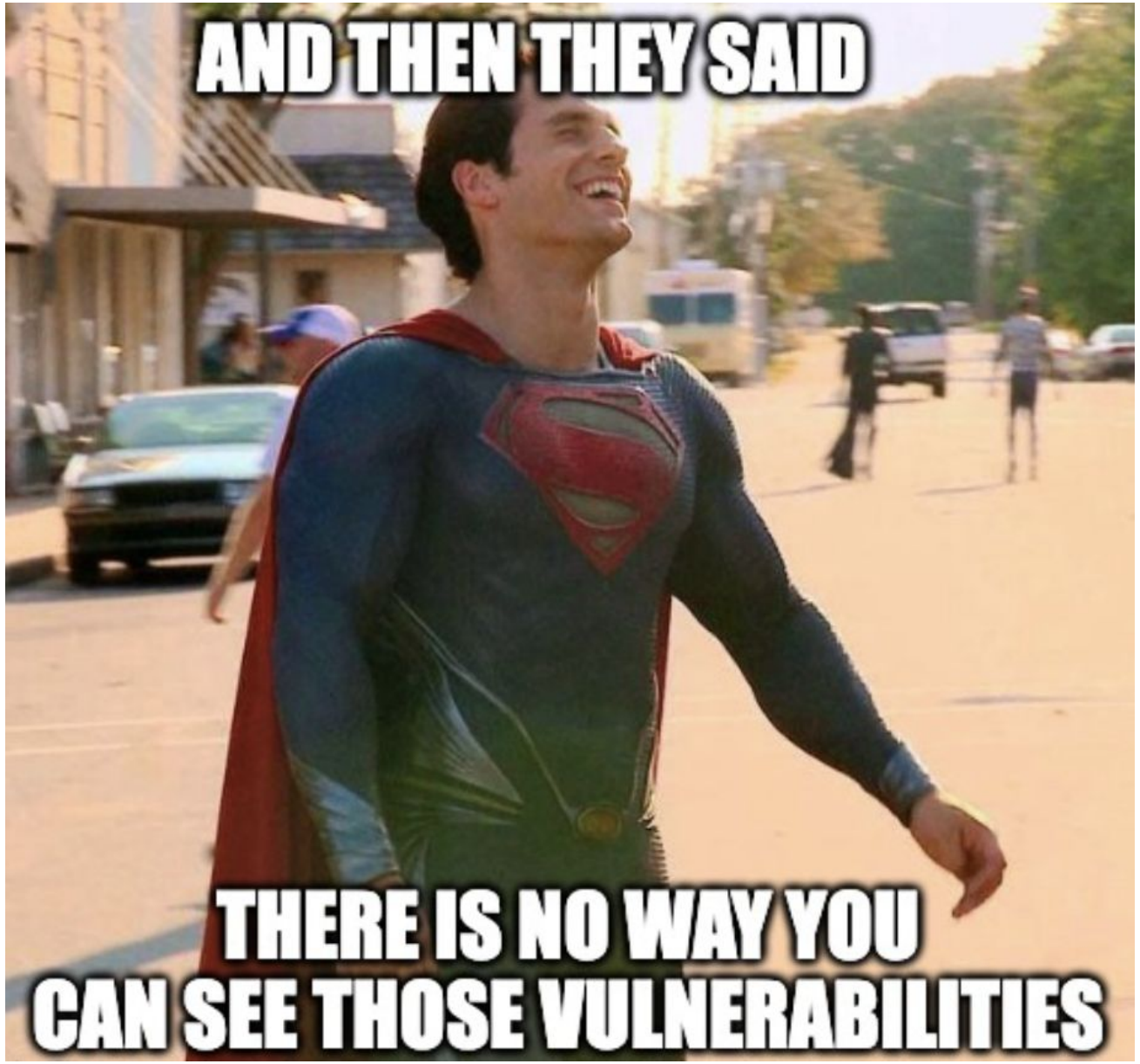
We now have a more complete SBOM!

KubeCon CloudNativeCon North America 2022

```
I ~/d/diesel_demo  *~ wget -q -O - ( \
                      rekor-cli get \
                      --uuid 24296fb24b8ad77ad9e011a61022dccf5d7fda6f827589d25b85eabebb721bf2f1004085b3b2402f | \
                      grep 'Attestation' | \
                      cut -f2 -w | \
                      jq -r .predicate.sboms[0].uri \
                      ) | grype
NAME              INSTALLED  FIXED-IN  TYPE        VULNERABILITY        SEVERITY
crossbeam-channel  0.3.9      0.4.4    rust-crate  GHSA-v5m7-53cv-f3hx  High
crossbeam-channel  0.3.9      0.4.3    rust-crate  GHSA-9g55-pg62-m8hh  High
crossbeam-utils    0.6.6      0.8.7    rust-crate  GHSA-qc84-gqf4-9926  High
crossbeam-utils    0.7.2      0.8.7    rust-crate  GHSA-qc84-gqf4-9926  High
failure            0.1.8               rust-crate  GHSA-jq66-xh47-j9f3  Critical
hyper              0.10.16             rust-crate  CVE-2021-32714       Critical
hyper              0.10.16             rust-crate  CVE-2020-35863       Critical
hyper              0.10.16             rust-crate  CVE-2021-32715       Medium
hyper              0.10.16    0.14.10  rust-crate  GHSA-f3pg-qwvg-p99c  Low
hyper              0.10.16    0.14.12  rust-crate  GHSA-f67m-9j94-qv9j  High
hyper              0.10.16    0.14.10  rust-crate  GHSA-5h46-h7hh-c6x9  Medium
metrics-util       0.5.0      0.7.0    rust-crate  GHSA-cwvc-87xq-pc5m  High
metrics-util       0.5.0      0.7.0    rust-crate  GHSA-3hxh-7jxm-59x4  Medium
owning_ref         0.4.1               rust-crate  GHSA-9qxh-258v-666c  Medium
tokio              0.1.22     1.8.4    rust-crate  GHSA-fg7r-2g4j-5cgr  High
traitobject        0.1.0               rust-crate  GHSA-pp8r-vv2j-9j5v  Critical
I ~/d/diesel_demo  *~
```

```
[base] 1:nvim  2:fish  3:fish  4:syft* 5:[tmux]-
I ~/d/diesel_demo  *~… syft -o json diesel_demo:1.0.0 | grype
 ✔ Loaded image
 ✔ Parsed image
 ✔ Cataloged packages      [14 packages]
No vulnerabilities found
```

New Super Powers are great



Caveats

- This uses the experimental Rekor attestation search function
  - This will be replaced by **GUAC** in the future
    (There's a talk on it tomorrow! https://sched.co/182Jr - Thursday 11am)

- Attestation Format is not current in-toto for compatibility reasons (uploads entire sbom which exceeds the size that rekor will store it)
  - Custom format is more of a pointer rather than a data store

- Currently, the trust policy defaults to anything sigstore/rekor includes
  - Future configuration on who to trust (e.g. list of emails for Fulcio OIDC authentication) - Issues #1159 and #1115

- If multiple rekor search results, take the most recent

Conclusions

- Start uploading software bill of materials to your releases that encapsulates the source that goes into your release artifact

- Make these documents discoverable via rekor

- Enable consumers of your software to see the entire picture

Links Slide

- Syft Tool: https://github.com/anchore/syft
- Rekor SBOM Uploader: https://github.com/lumjjb/sbom-attest
- Sigstore: https://www.sigstore.dev/