
Security of the Software Supply Chain
with
sigstore
Hervé Boutemy & Louis Jacomet
Maven™
Gradle Build Tool
sonatype

AGENDA

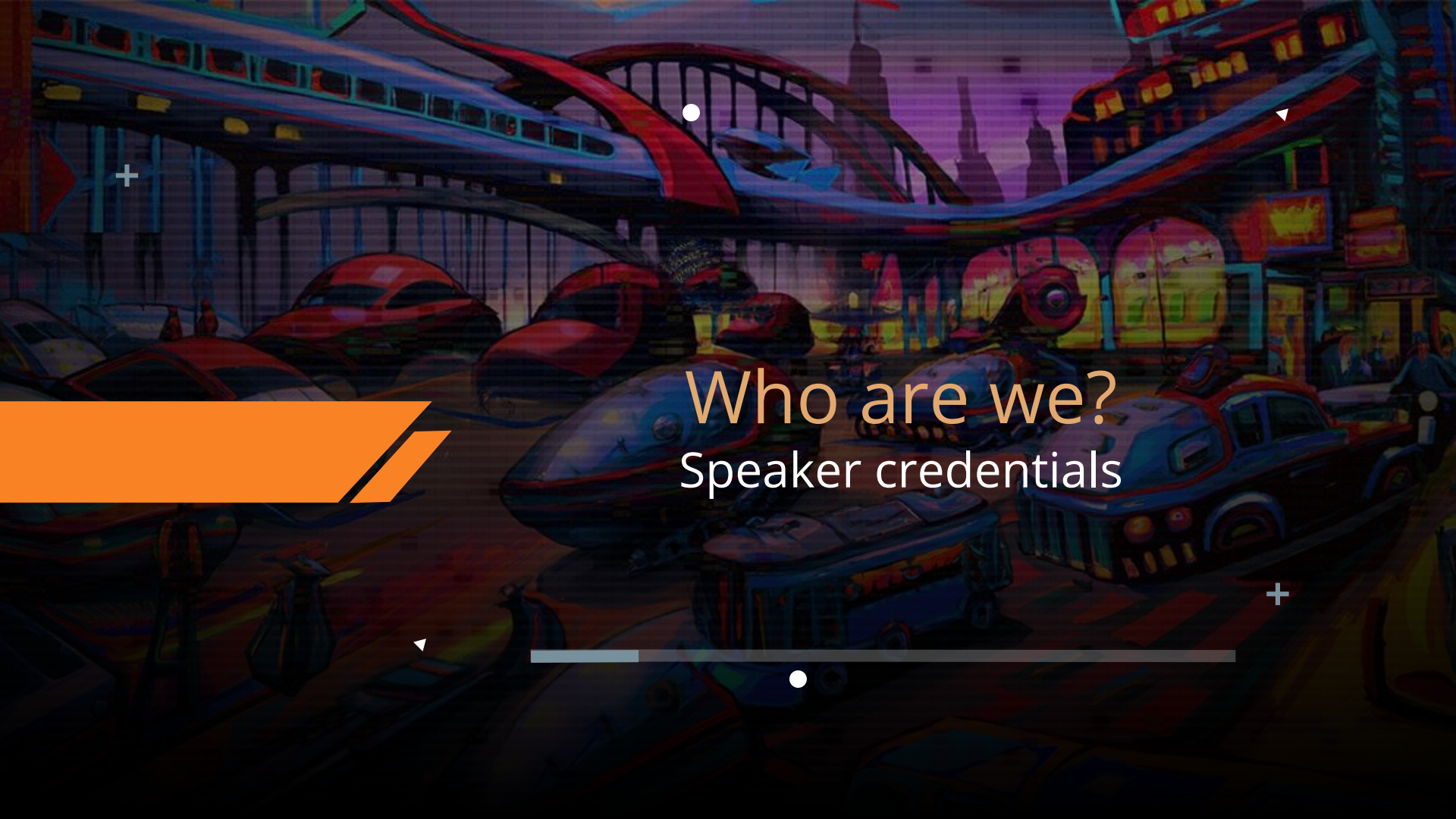
Software Supply Chain security
Concepts, risks and attacks

Sigstore
Keyless, traceable

Publish with Sigstore
Support in Maven Central

Sign with Sigstore
Plugins for Gradle and Maven

Verify a Sigstore signature
Challenges for Gradle and Maven



Who are we?

Speaker credentials

Hervé Boutemy

sonatype

```
speaker {
   company = "Sonatype"
   joined = 2019
   position = "Solutions Architect and more ..."
   past = listOf(
      "French Banks" from 1996 to 2018,
      "Java 'Hello, World!' Applet" in 1996
   )
   social = “@hboutemy"
   github = "hboutemy"
}
```

Hervé Boutemy

```
"Apache Software Foundation" {
   projects = listOf(
      "Maven" from 2005,
      "Community Development" from 2014,
      "Attic" from 2016
   )
   past = listOf(
      "Maven PMC Chair" from 2014 to 2016
   )
   member = 2011
   email = "hboutemy@apache.org"
}
"Reproducible Builds" {
   since = 2017
}
"OWASP CycloneDX" {
   since = 2023
}
```

THE APACHE SOFTWARE FOUNDATION

Maven

Reproducible Builds

OWASP CycloneDX

Gradle 💖 Maven

Louis Jacomet

```
speaker {
   company = "Gradle"
   joined = 2018
   position = "Support Team Lead and more ..."
   previously = "Dependency Management, JVM plugins"
   past = listOf(
      "Terracotta / Ehcache" in 2013,
      "Devoxx Belgium Committee" in 2012,
      "Contractor" in 2002,
      "Java 'Hello, World!'" in 1997
   )
   failures = generateSequence(code) { bugs }
   social = listOf("@ljacomet@foojay.social", "@ljacomet")
   github = "ljacomet"
   web = "https://jacomet.dev"
   extra = "Not fully figured out how to stay out of management !?!"
}
```

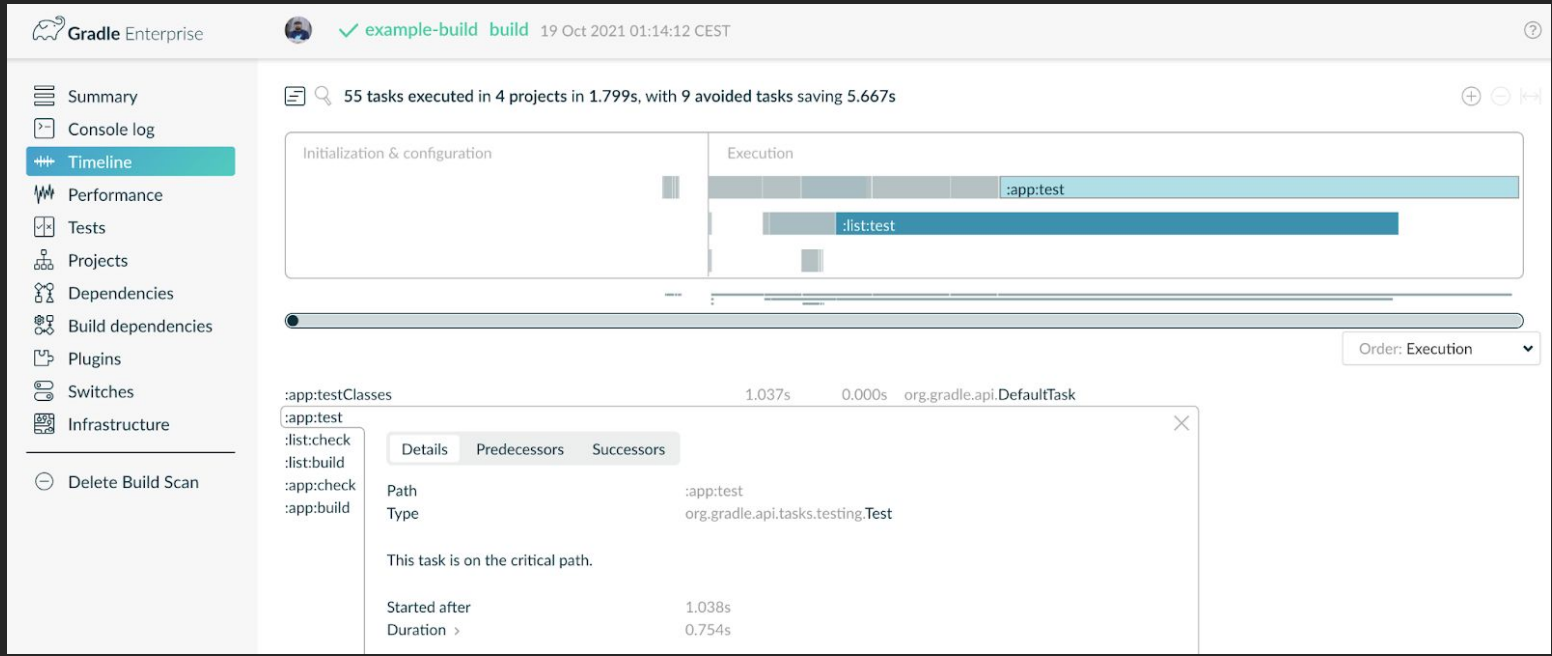
+

Gradle Build Tool

- Apache 2.0 licensed build tool
- JVM based
- Kotlin and Groovy configuration DSL
- 30+ millions downloads / month
- Extensive plugin eco-system

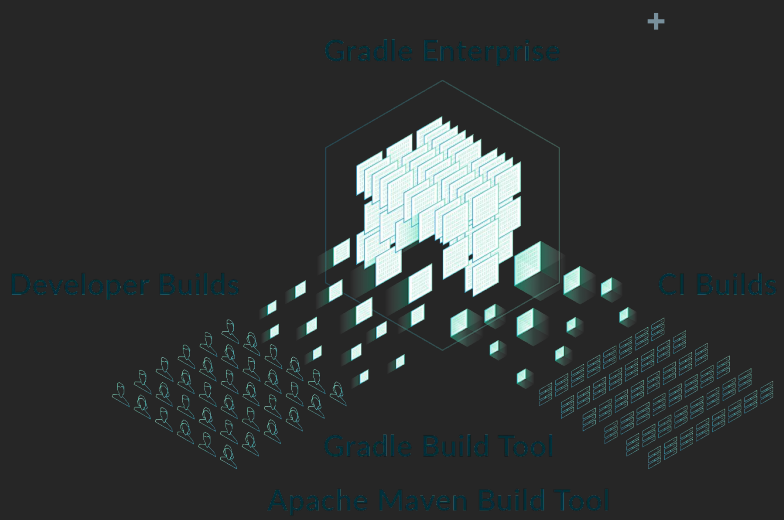
Gradle Inc.

Build scans



Gradle Inc.

Gradle Enterprise



Developer Productivity Engineering

Software Supply Chain security

Concepts, risks and attacks

Concepts

“hackers don't attack individual devices or networks directly, but rather the companies that distribute the code used by their targets”

https://www.wired.com/story/supply-chain-hackers-videogames-asus-ccleaner/#

Concepts

Supply chain attack

Infect a tool or a library

Used by another tool or library

With the goal of getting it distributed
executed

In a trusted way

Attacks

- CC Cleaner / Asus in 2018
  - (Auto) update system hacked
  - Update installed a backdoor
  - 2.27 millions downloads
  - 1.65 millions phone home
  - 40 follow ups, high targets only

https://www.wired.com/story/inside-the-unnerving-supply-chain-attack-that-corrupted-ccleaner/

Attacks

- Solar Winds in 2020
  - System management software
  - Backdoor inserted in Orion system
  - Software updates propagated the hack from March to December
  - Backdoor allowed to reach customers of customers

https://www.techtarget.com/whatis/feature/SolarWinds-hack-explained-Everything-you-need-to-know?amp=1

Attacks

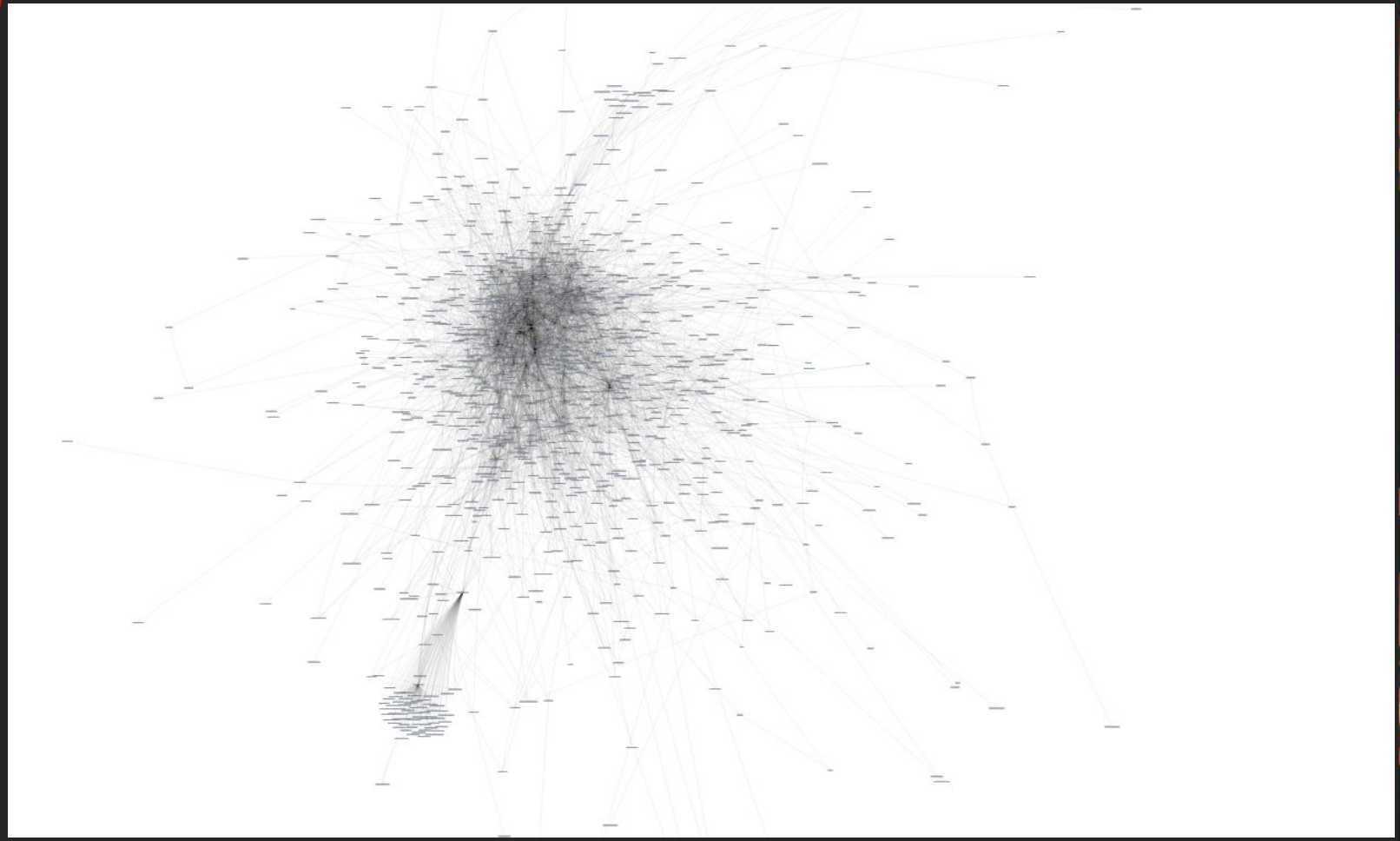
- Infected Gradle wrapper in January 2023
  - Part of a first time contributor PR
  - Extracts and posts Discord credentials
    - When running the build
  - Replaces a project dependency
    - Injecting code that
    - Opens up a command and control endpoint at execution

https://blog.gradle.org/wrapper-attack-report

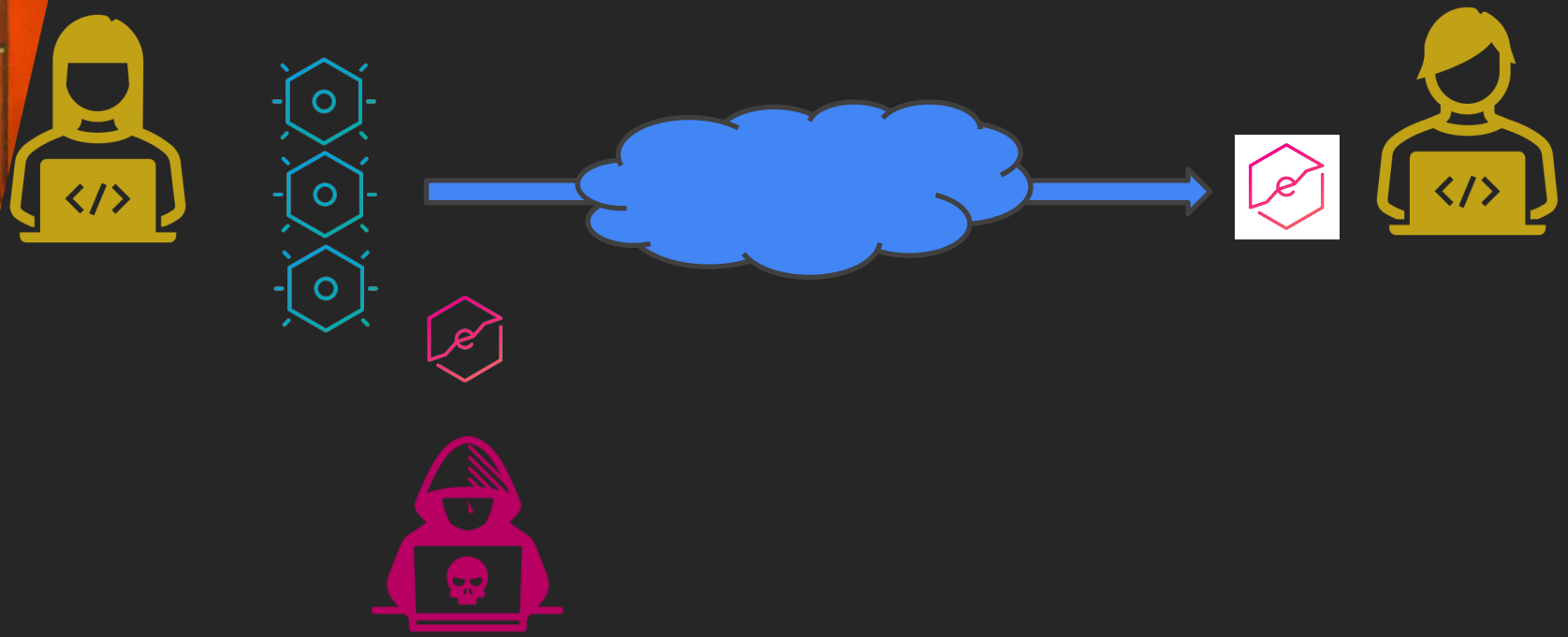
Risks

- Supply chain attacks are a reality, not just a concept
- Developers are the target audience
- Incredible scale out effect
- Some issued by Nation State actors

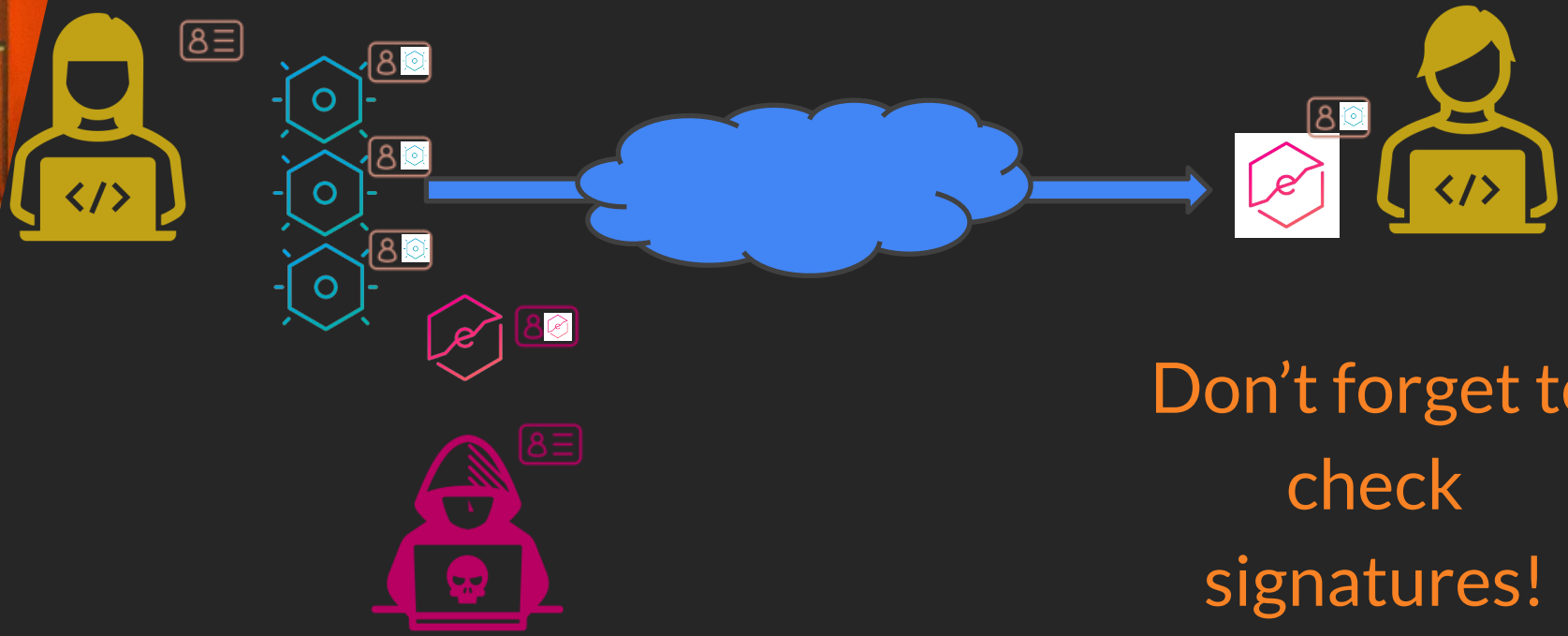
Risks



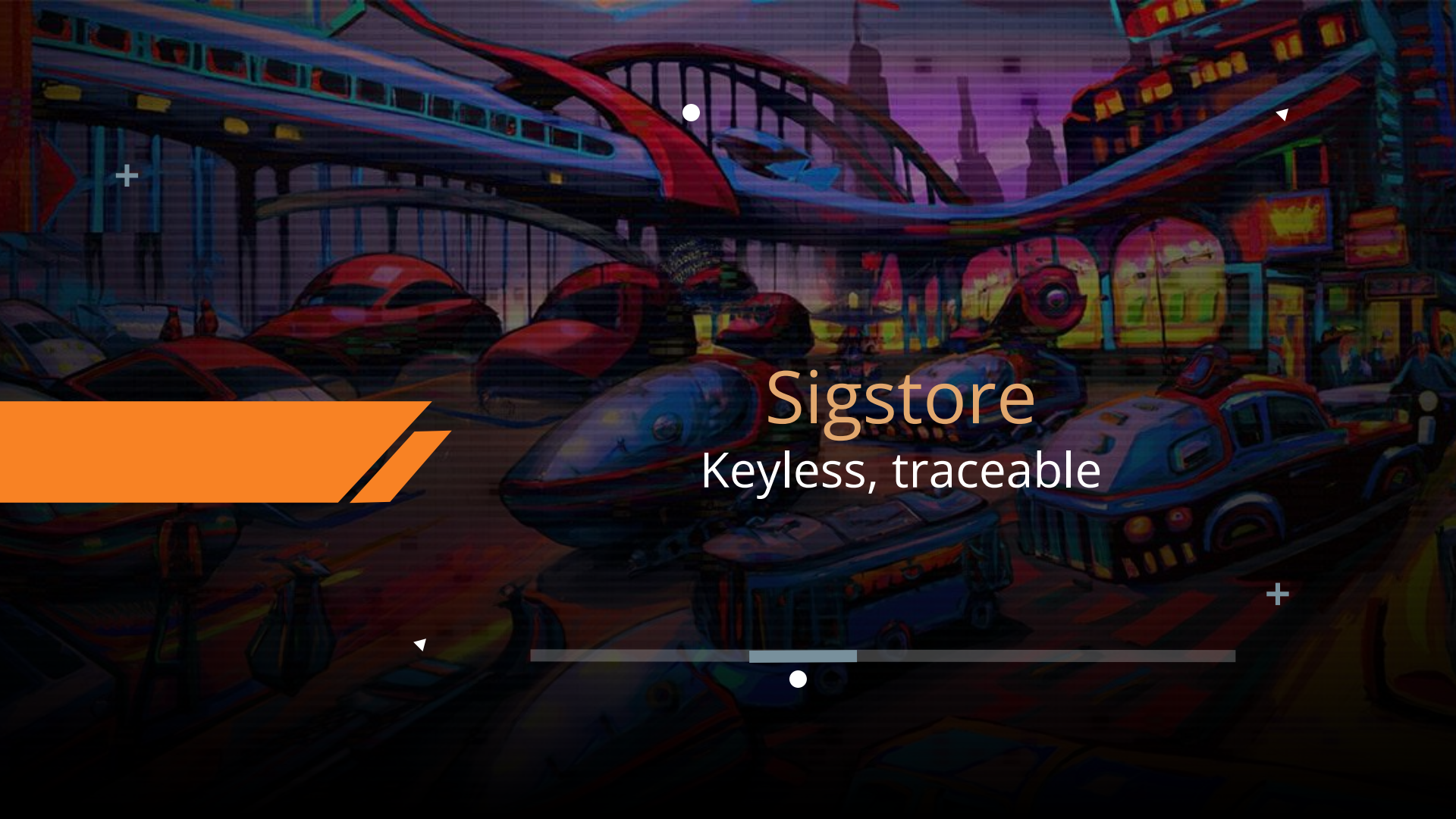
Attacks on Software Supply Chain



Signature for Software Supply Chain Protection



Don’t forget to check signatures!


Sigstore
Keyless, traceable

Sigstore

sigstore project: easy “keyless” signature

OpenSSF
OPEN SOURCE SECURITY FOUNDATION

- started mid-2020
- March 2021: The Linux Foundation project by Red Hat, Google and Purdue University
  - now more than 20 organizations (Chainguard, VMware, Twitter, Citi, Charm, Anchore, Iron Bank…)
- July 28,2021: cosign 1.0 release
  = container signing, verification and storage in registry

sigstore
cosign

- “Keyless” feature = work in progress
  Container first: OCI images + K8S
  involve software repositories + package managers/build tools actors:
  - Maven Central, Ruby Central, PyPI, npmjs, …
  - Maven, Gradle, RubyGems, pip, npm, …

"keyless"

sigstore « keyless »

- CLI tools
  - cosign: https://github.com/sigstore/cosign
  - gitsign: https://github.com/sigstore/gitsign

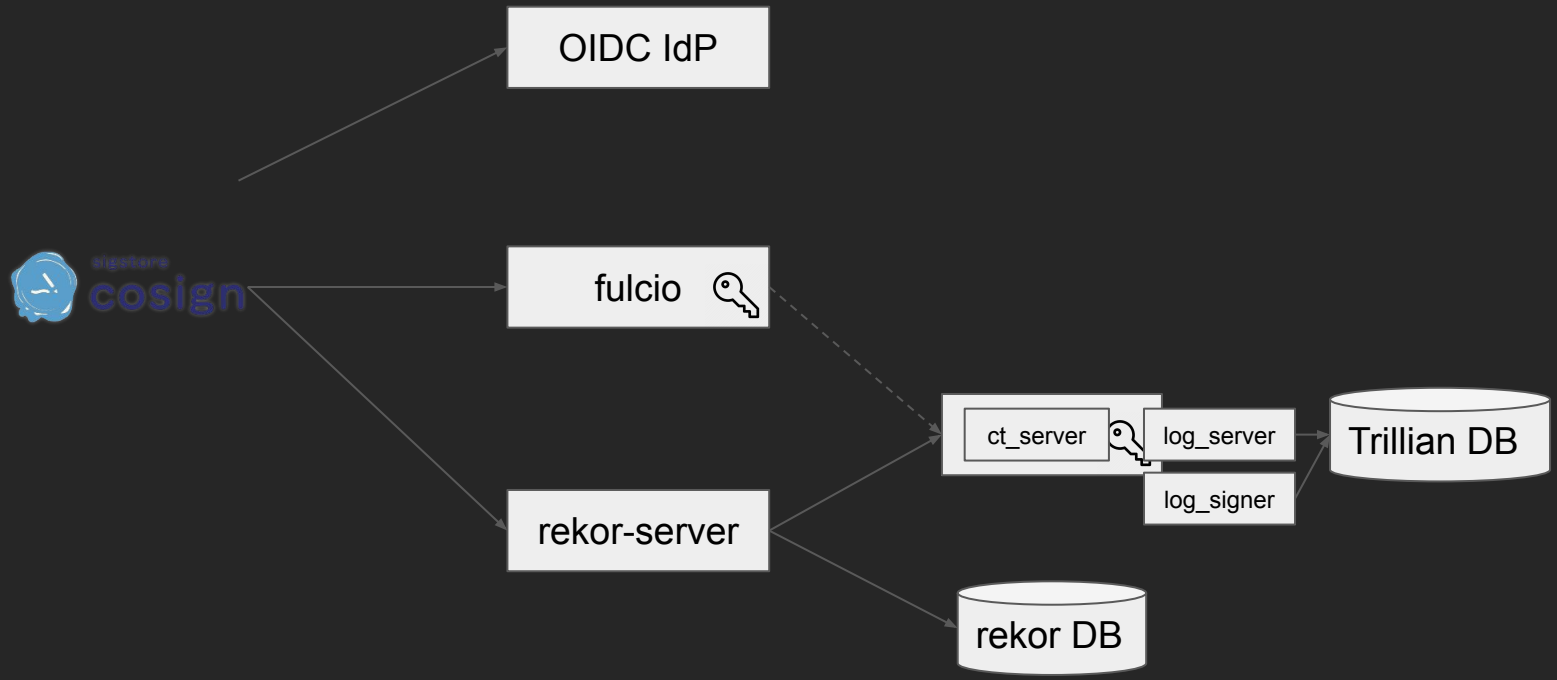
- documentation, libs for many languages, …

- server software

  - fulcio: https://github.com/sigstore/fulcio
    Root-CA for short-lived code signing certs,
    issuing certificates based on an OIDC identity provider

  - rekor: https://github.com/sigstore/rekor
    Transparency log

Fulcio & Rekor architecture



Community Services

sigstore shared community services

public services (prod + sigstage):

https://fulcio.sigstore.dev/

https://oauth2.sigstore.dev/auth = dex

https://rekor.sigstore.dev/

https://ctfe.sigstore.dev/

Root keys (TUF): https://github.com/sigstore/root-signing

objectives:
- community managed
- 99.5% uptime SLO: https://status.sigstore.dev/
- diversity of content monitors

Workflow

“keyless” signing workflow

- generate a session keypair
- sign files

- contact Fulcio to get short-lived certificate
  - uses OIDC IdP
  - starts a 10-minutes timespan

- register signatures to Rekor

- (optional) save results to sigstore bundle file


DEMO
“keyless” signing

“keyless” signing workflow demo

demo

https://github.com/hboutemy/sigstore-java-poc


Break
See you in 15 minutes


Publish with Sigstore
Support in Maven Central

Early 2000...

Early Maven Central: Maven 1 & 2.0...

- Apache Software Foundation:
  - reference release archives on an ASF disk: permissions managed by Unix groups
  - download content from public mirrors
  - signature policy since 2002
    - GPG sign source code releases
    - get signature (.asc) from ASF servers
    - check signature against KEYS file in Apache distribution area for each project

- Maven Central:
  - Apache Maven 1.x => Maven 1 repository format: short groupId, no signature
    « Maven Central 1 » = http://repo1.maven.org/maven/ (now obsolete)
  - Apache Maven 2.x and more => Maven 2 repository format
    structural changes: reverse DNS groupId as directory, POM v4
    Maven Central = https://repo1.maven.org/maven2/
  - Reference content on a disk managed by Maven team
    - Synchronisation shell scripts to ingest content from other organizations
  - Network of public mirrors
  - PGP signature support: .asc files
  - PGP signature mandatory since 2006 in Maven Central

Maven Central

High Level Architecture

sonatype

Maven Central by the Numbers

In 2021, developers around the world made more than

496 BILLION

requests from Maven Central.

Maven Central by the Numbers

Statistics as of 6 May 2022

8.8m

component versions stored in ...

27TB

... of files representing approximately ...

79k

... namespaces / organizations / publishers

Modern Architecture?

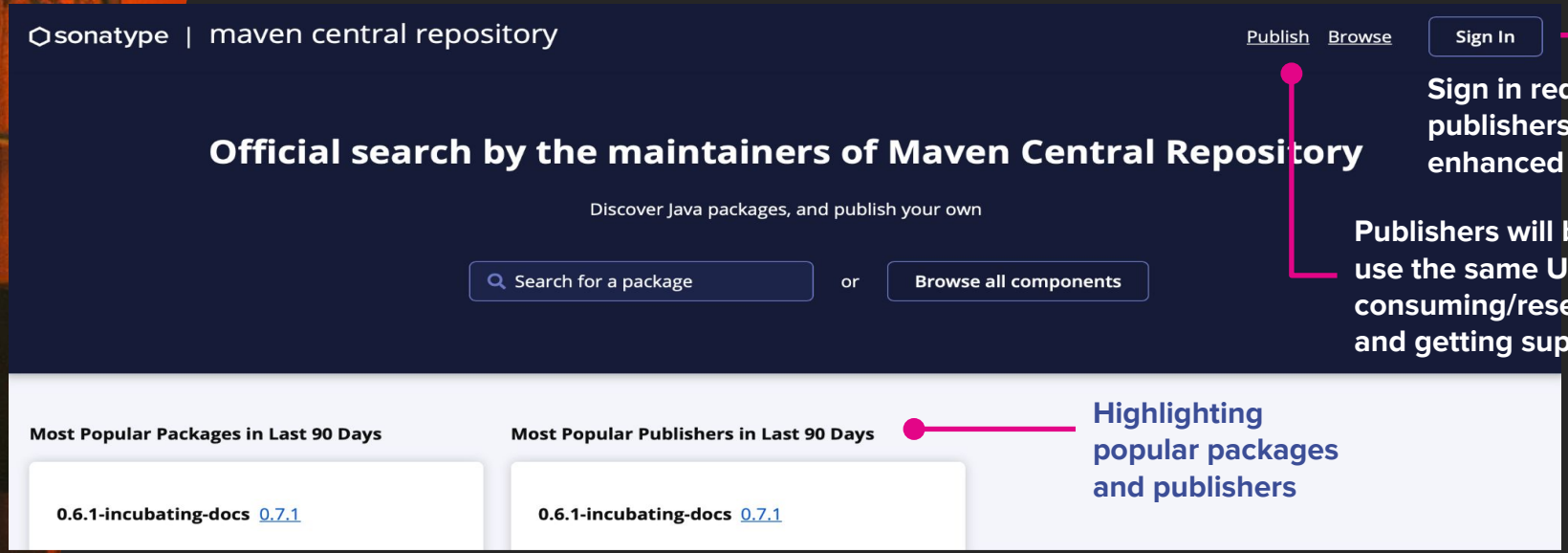
sonatype | Maven Central Repository Search Quick Stats GitHub

Search

Advanced Options | API Guide

Official search by the maintainers of Maven Central Repository

https://central.sonatype.com/



Sign in required for publishers and enhanced metadata.

Publishers will be able to use the same UI for consuming/researching and getting support

Highlighting popular packages and publishers

Devoxx France

2023

Maven Central

Adding Sigstore support

Layout

Maven Central Layout

- GAV directory: `${groupId dir}/${artifactId}/${baseVersion}`
  - 1 GAV per Maven module, 1 release = 1 to > 1 000 GAV
- Artifact files: `${artifactId}-${version}-${classifier}.${extension}`
  - Minimum:
    - .pom
    - .jar
    - -sources.jar
    - -javadoc.jar
  - Control files:
    - .md5
    - .sha1
    - .asc
    - .sha256
    - .sha512
- 1 typical GAV = 4 artifact files * 5 control files = 20 files

- Typical waste: .asc.sha* => 5 + 4 = 9 control files

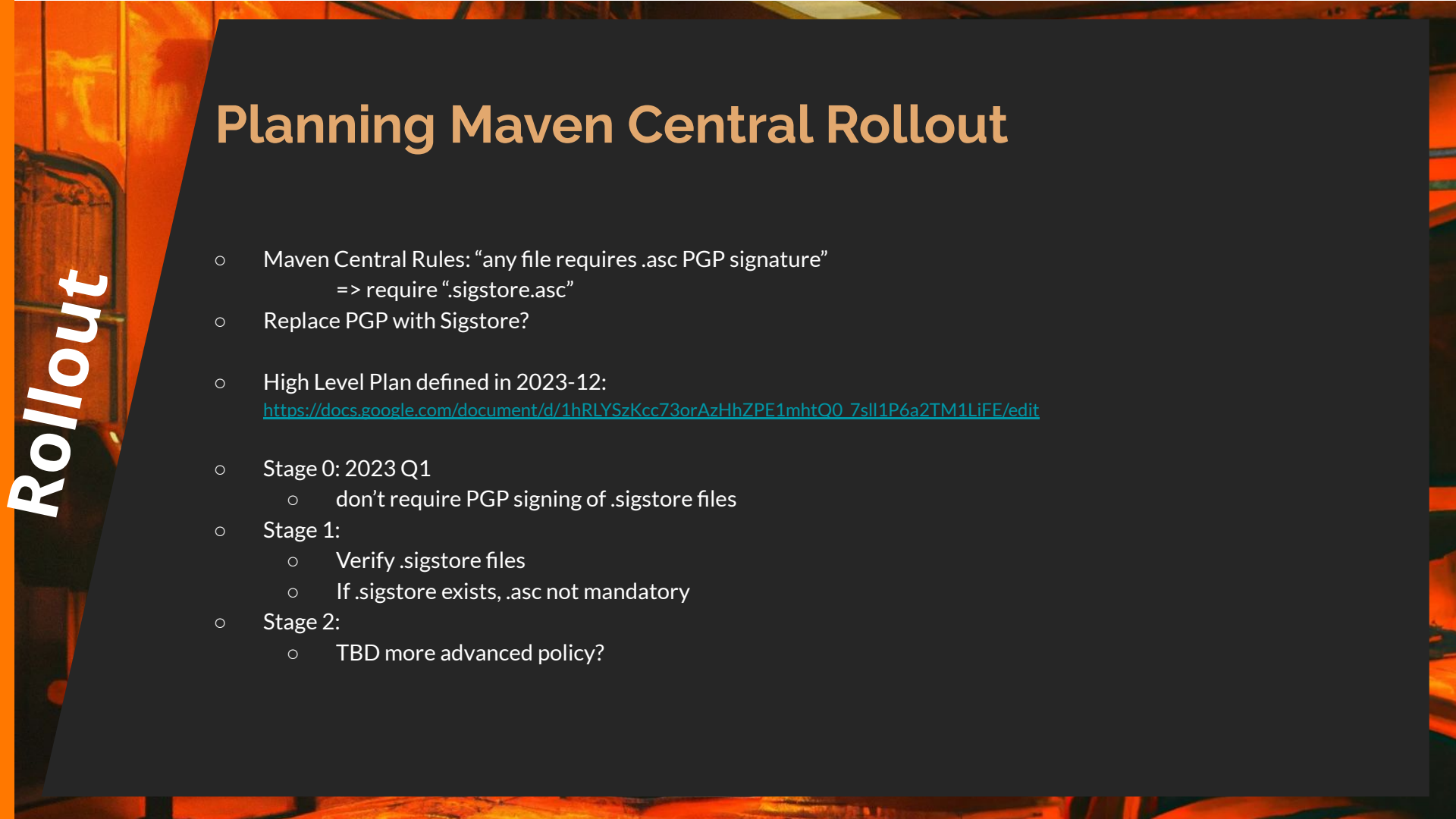
Adding Sigstore offline Signature

- 2022-10 working group: Additional control files:
  - .pem
  - .sig
  - .bundle

  => 5 + 3 = 8 control files
  - if .asc & .sha* & .asc.sha* waste: 5 + 3 * (5 to 9) = 20 to 32 control files
- Exemple: https://repo.maven.apache.org/maven2/dev/sigstore/sigstore-java/0.1.0/

- “Sigstore bundle format”:
  - .sigstore

  => 5 + 1 = 6 control files
- Sigstore-wide project: https://github.com/sigstore/protobuf-specs
  - format defined 2023-01
  - Implemented in sigstore-java 2023-02
- Exemple: https://repo.maven.apache.org/maven2/dev/sigstore/sigstore-java/0.4.0/



Planning Maven Central Rollout

- Maven Central Rules: “any file requires .asc PGP signature”
  => require “.sigstore.asc”
- Replace PGP with Sigstore?

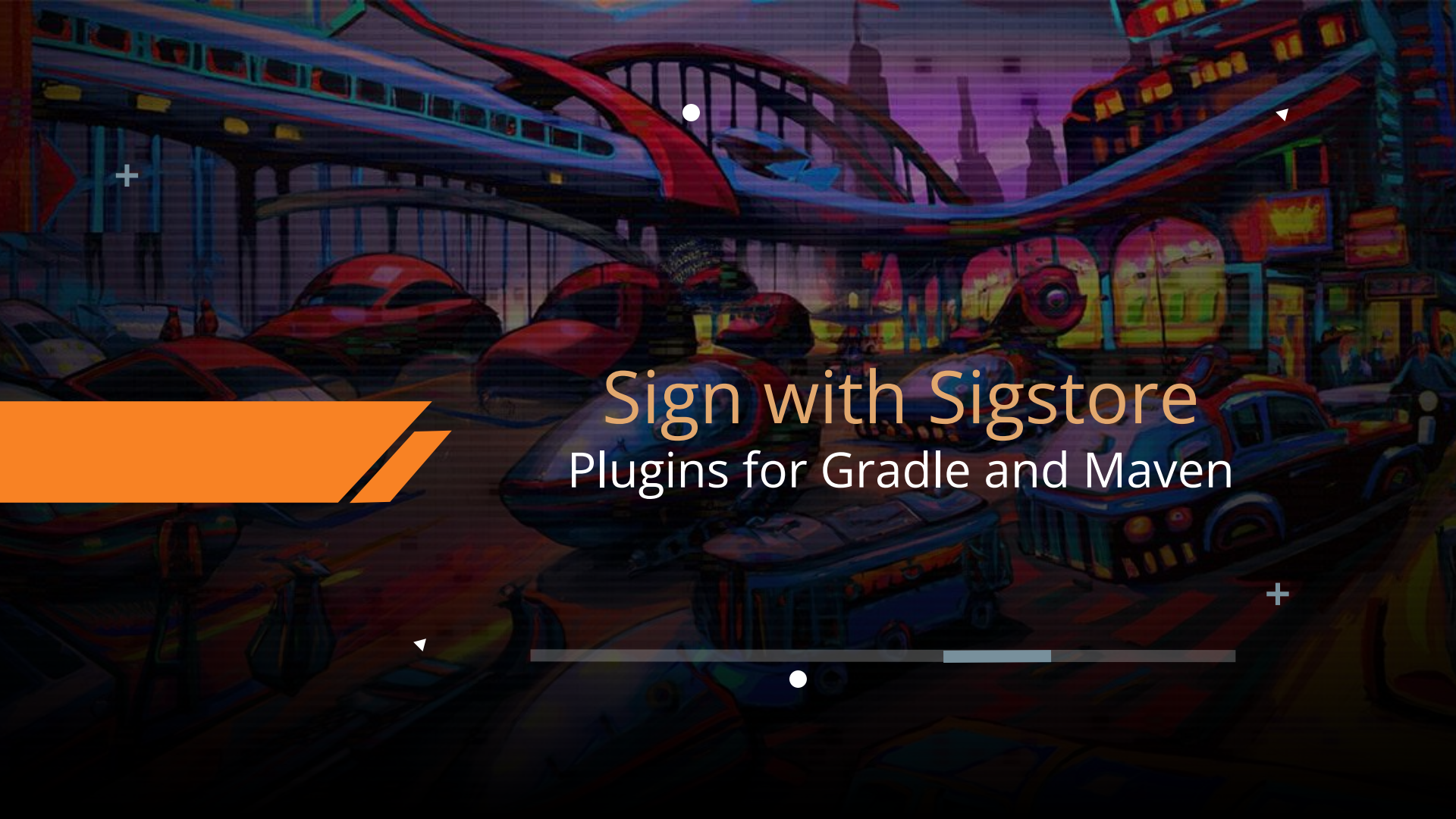
- High Level Plan defined in 2023-12:
  https://docs.google.com/document/d/1hRLYSzKcc73orAzHhZPE1mhtQ0_7sll1P6a2TM1LiFE/edit

- Stage 0: 2023 Q1
  - don’t require PGP signing of .sigstore files
- Stage 1:
  - Verify .sigstore files
  - If .sigstore exists, .asc not mandatory
- Stage 2:
  - TBD more advanced policy?

Bundle

- Sigstore bundle format
  - Single file for each signed artifact
  - Regroups information otherwise found in multiple files
- Content
  - Sha256 digest of signed artifact
  - Artifact signature
  - Fulcio certificate chain
  - (Optional) Entry in Rekor transparency log


DEMO
Sigstore bundle verification


Sign with Sigstore
Plugins for Gradle and Maven

Devoxx France

2023

Sigstore signing

Signing with Maven

Maven sign

- Many plugins written in the past at various stages with various learnings:
  - Sigstore-maven-plugin (2021-03 to 2022-03)
    https://github.com/sigstore/sigstore-maven-plugin/
    = jarsigner: embedded signature in jar
  - Sigstore-maven (2022-05 to 2022-09)
    https://github.com/sigstore/sigstore-maven
    = no sigstore-java, no bundle
- New try using sigstore-java:
  https://github.com/apache/maven-gpg-plugin/pull/43


DEMO
Maven Sigstore signing

Learnings

- Next sigstore java meeting is today at 17:30
- How to write integration tests?
- Sigstore session spans: reauthentication?
  - multi-module?
  - multi-module > 10 minutes?
- Workaround:

```
mvn deploy
-DaltDeploymentRepository=local::default::file:./target/staging-deploy
```

- Longer term: Maven core “run at end”?

Devoxx France

2023

Sigstore signing

Signing with the Gradle Sigstore plugin


DEMO
Gradle Sigstore signing

Gradle sign

- Plugin `dev.sigstore.sign`
  - Version `0.4.2` at this time
  - Requires Java 11 (from `sigstore-java`)
  - Requires Gradle 7.5
  - Signs all publications by convention
- Plugin `dev.sigstore.sign-base`
  - No signing convention
- See plugin documentation for details

Gradle sign

- If using `signing` as well
  - Need custom code to **not** GPG sign `.sigstore` files
- Proper integration will require new APIs in Gradle
  - Separate publication artifacts from derived artifacts

Devoxx France

2023

Sigstore signing

Non-interactive signing


DEMO
Gradle Sigstore signing
on GitHub actions

Gradle sign

- OpenID Connect support in GitHub actions
  - Requires specific permissions

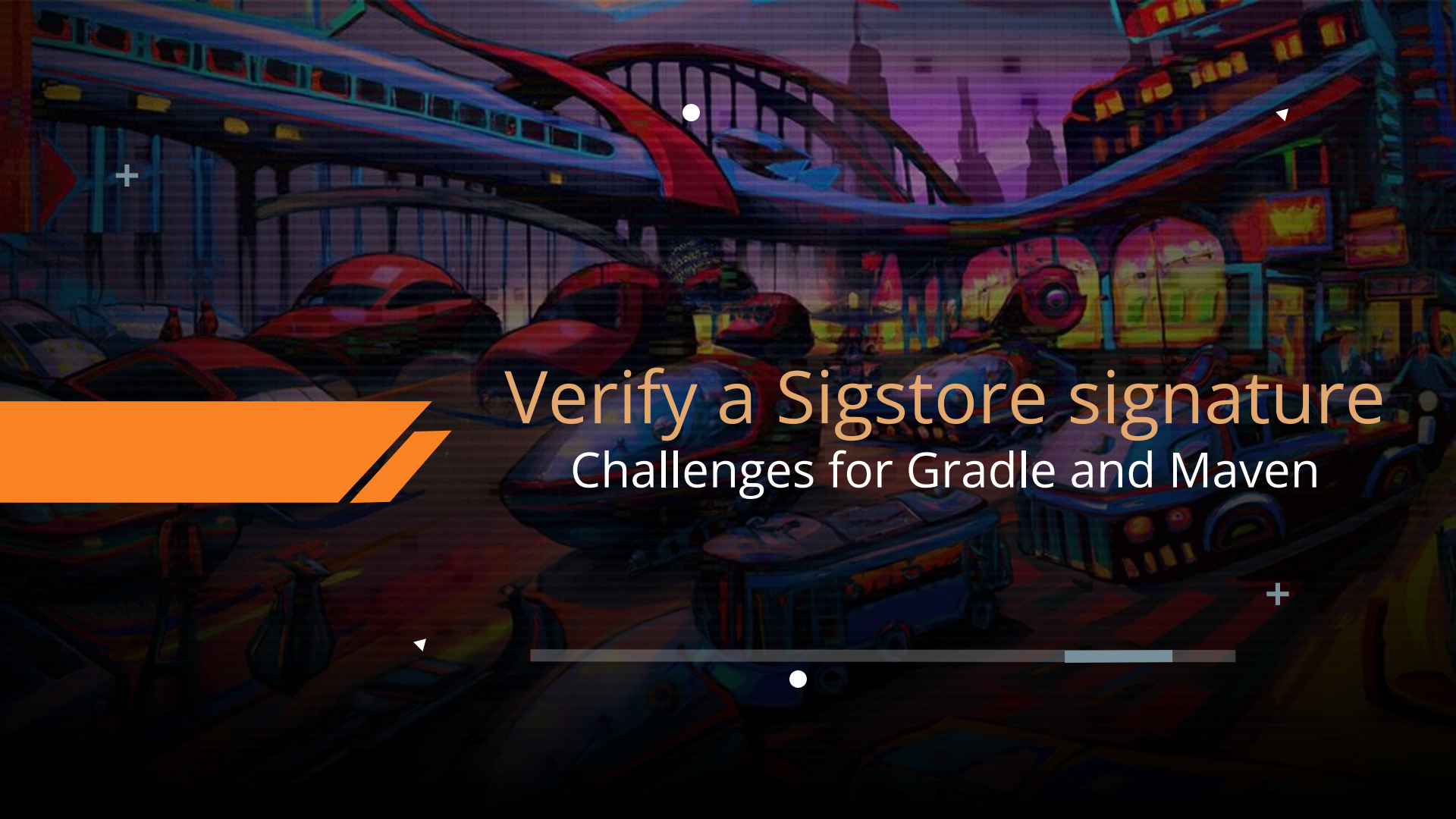
```
permissions:
  id-token: write
```

  - Leverages machine identifiers

"https://github.com/ljacomet/logging-capabilities/.github/workflows/gradle-publish.yml@refs/heads/sigstore-tests"

sigstore-java

- Supports a limited set of OIDC clients
  - Web client with 
  - GitHub actions
- Other clients?
  - Support needs to be added


Verify a Sigstore signature
Challenges for Gradle and Maven

Devoxx France

2023

Sigstore verification

Why verification matters?

Verify artifacts

- Dependencies from repositories
  - MD5 checksums are no longer safe
  - SHA1 are not much better
  - PGP signatures are often not verified
- Checksums are for download integrity
  - When cross referencing sources
- Signatures are for provenance
  - Malicious authors can sign too

Verify artifacts

- Gradle supports dependency verification
  - Signature or checksum based
  - All information contained in a file in the build

```
<trusted-keys>
    <trusted-key id="019082bc00e0324e2aef4cf00d3b328562a119a7" group="org.openjdk.jmh"/>
    <trusted-key id="0785b3eff60b1b1bea94e0bb7c25280eae63ebe5" group="org.apache.httpcomponents"/>
    <trusted-key id="07e20f0103d9dfc697c490d0368557390486f2c5" group="org.awaitility"/>
    <trusted-key id="08f0aab4d0c1a4bdde340765b341ddb020fcb6ab" group="org.bouncycastle"/>
```

```
<component group="aopalliance" name="aopalliance" version="1.0">
    <artifact name="aopalliance-1.0.jar">
        <sha256 value="0addec670fedcd3f113c5c8091d783280d23f75e3acb841b61a9cdb079376a08" origin="Verified" reason="Artifact is not signed"/>
    </artifact>
</component>
<component group="cglib" name="cglib-nodep" version="2.1_3">
    <artifact name="cglib-nodep-2.1_3.jar">
        <sha256 value="e77f0b091a800acd15217bbc3c3629c1af0925b55b271fc5d0586422985639bb" origin="Verified" reason="Artifact is not signed"/>
    </artifact>
</component>
```

Verify artifacts

- Maven has plugins for dependency verification
  - Verification of plugins
  - Verification of metadata files
  - Uses a properties file like format for configuration
  - Only for PGP verification, not checksums

https://www.simplify4u.org/pgpverify-maven-plugin/

Devoxx France

2023

Sigstore verification

Verify what?

Verify sigstore

- ➔ Verification of valid Sigstore bundle
  - ◆ Chain of trust for Fulcio
  - ◆ Valid signature of digest
  - ◆ Valid Rekor entry
    - Online vs. offline

- ➔ Same guarantees as PGP
  - ◆ Bundle is cryptographically valid

Verify sigstore

- OpenID Connect enables more
  - Email as identity
  - From trusted identity provider

- But what about machines?
  - From trusted identity provider
  - ?? as identity

Verify sigstore

- Elements of ownership proof
  - One OpenID provider
  - Combined with an identity
    - Email for humans
    - Machine ID for automation
    - Maybe support wildcards
  - And multiple of the above

- This might evolve based on Sigstore usage and release patterns

Devoxx France

2023

Sigstore verification

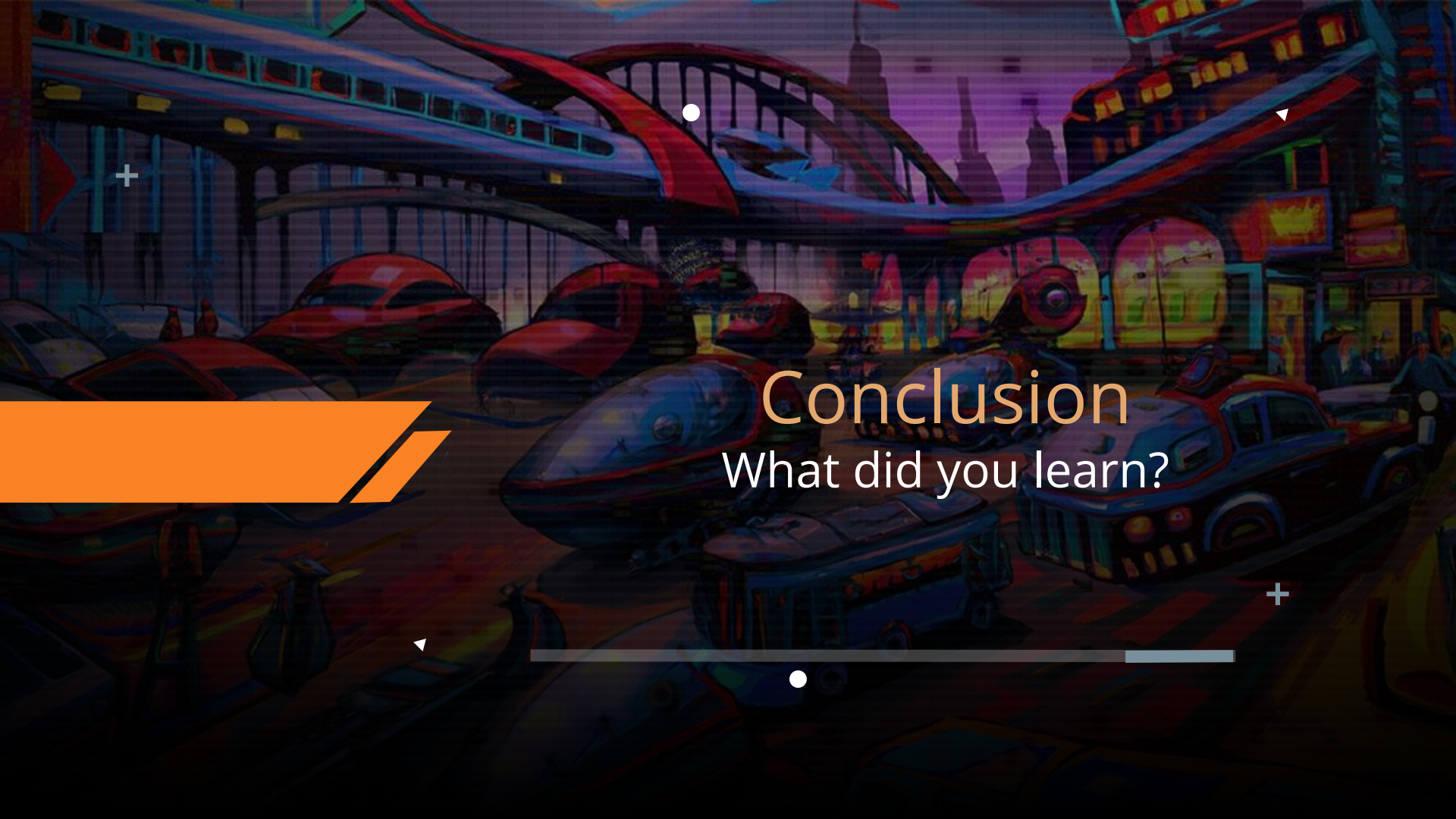
Support in Gradle and Maven

Verify sigstore in Gradle

- Path to adding sigstore verification in Gradle
  - Design extensions to the format
  - Add Sigstore support
    - Minor issue with mandatory Java 11 to run Sigstore
  - (Optional) Evolve `sigstore-java` to ease Gradle integration
    - Provide bundle verification APIs allowing wildcard matches
- No ETA

Verify sigstore in Maven

- TBD



Conclusion

What did you learn?

Takeaways

- Signing user experience is improved
- Tooling integration has challenges
  - Short lived keys
  - Vs. long running builds
- Supporting multiple workflows
  - Work in progress
  - Requires community contributions

Takeaways

- Verification remains a challenge
  - What to trust?
  - Where to find the information?

+

- Different “Sigstore” instances
  - *.sigstore.dev (& *.sigstage.dev)
  - Company ones: *.sigstore.mycompany.com
    - Public or private?

Takeaways

- Ownership information
  - Where can you find who can publish a library?
  - As a maintainer, document this
- Central ownership repository?
  - Breaking it means breaking the chain
- What about other ecosystems?
  - Npmjs, PyPI, Nuget gallery, ...


Thank you
for your attention