DEVOPS
FAST FORWARD

SECURING YOUR SUPPLY CHAIN

JFrog

AGENDA

- Introduction
- Attacker’s point of view
- Defense point of view
- Practical point of view

William Manning
Solutions Architect

@williammanning

Asaf Cohen
Dir. Security Solutions

@asaf.cohen

JFrog
Liquid Software Company

SOFTWARE SUPPLY CHAIN ATTACKS

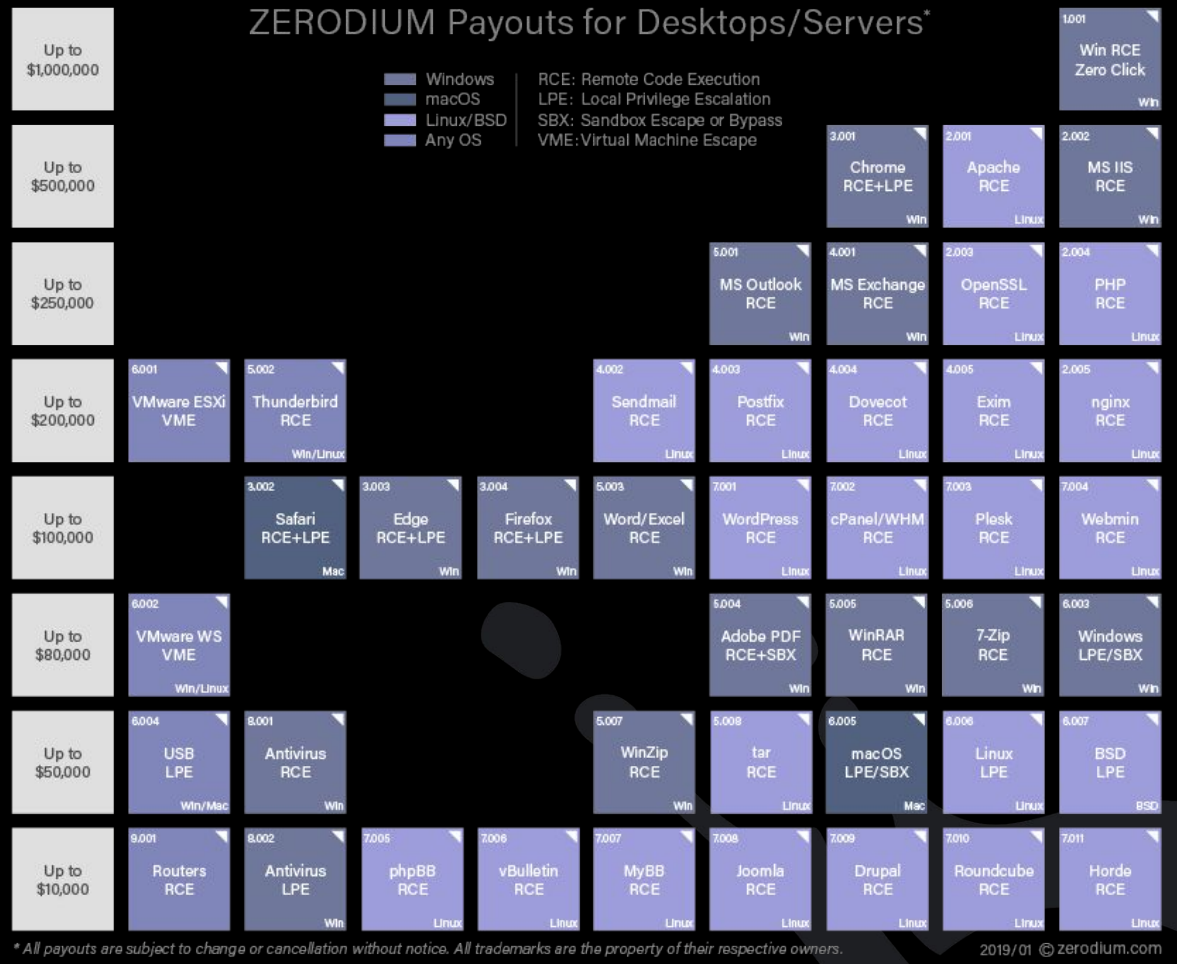
The definition we all know:

A technique in which an adversary slips malicious code or even a malicious component into a trusted piece of software or hardware.

SOFTWARE SUPPLY CHAIN ATTACKS

Why would an attacker go for this approach?

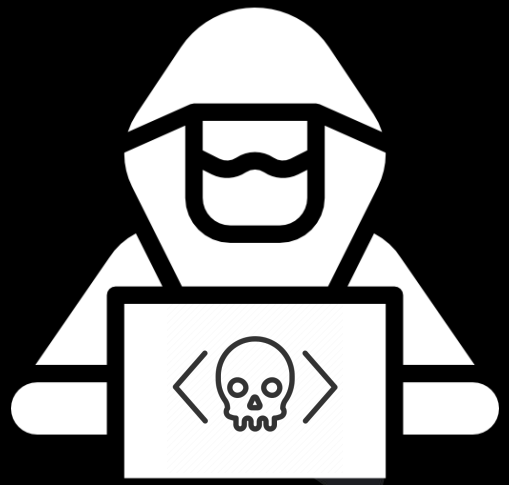
1. Low effort
2. Low technical skills required
3. High spread attack
4. Abuse the trust relationship between companies



SOFTWARE SUPPLY CHAIN ATTACKS

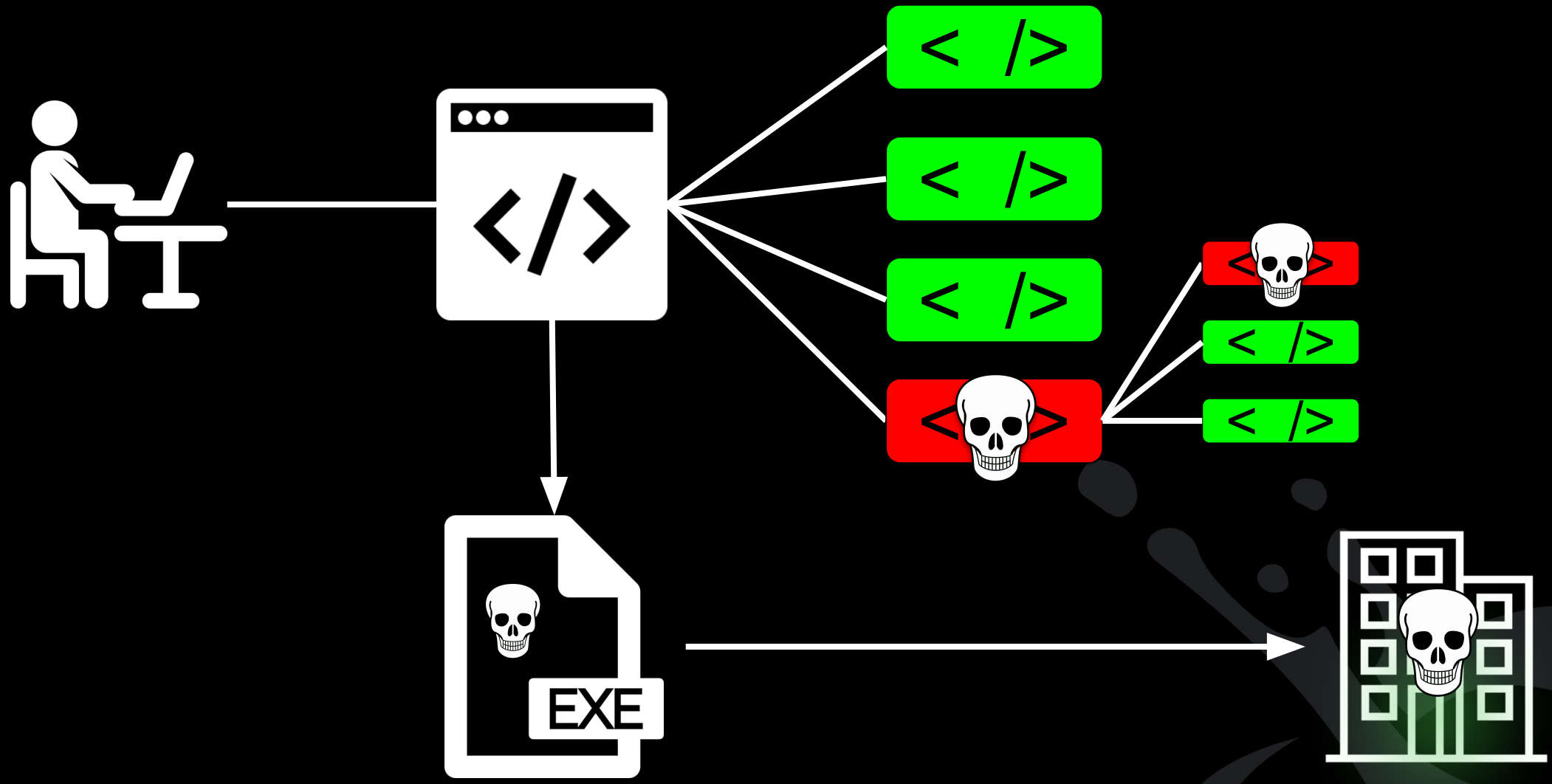
5. Attackers can code and blend into the “community”

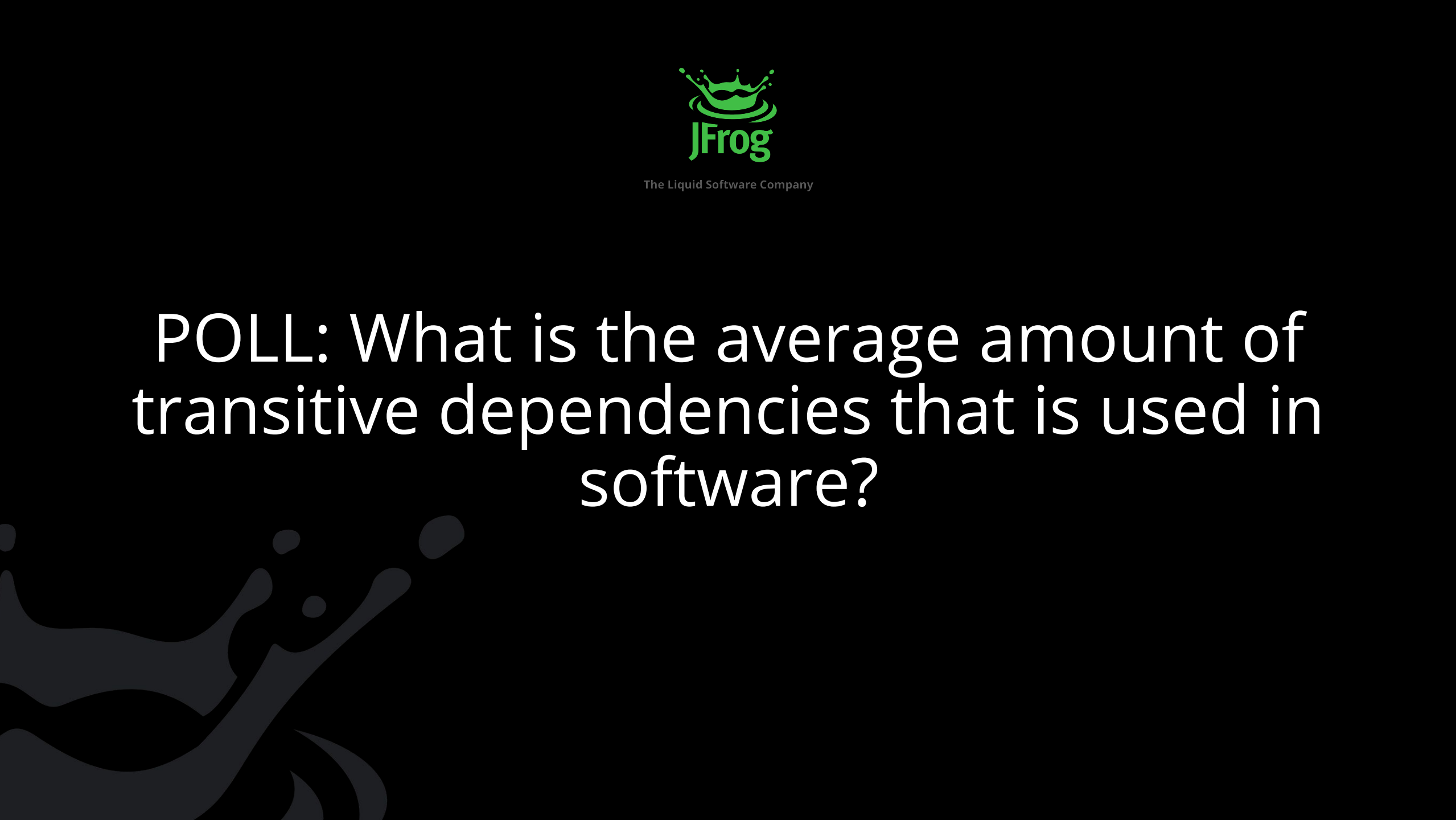
Backdoor



Malicious code

HOW DO ATTACKS OCCUR?





POLL: What is the average amount of transitive dependencies that is used in software?

The Liquid Software Company

In an average application, 85-90 percent of the codebase was open source.

99 percent of codebases contain at least some open source code and 75 percent used at least one vulnerable open source component.

The Liquid Software Company

49 percent of codebases it analyzed had at least one component with a high-risk vulnerability.

90 percent of applications used at least one open source component that was out-of-date by four or more years, or was abandoned

The Liquid Software Company

74 percent, of the applications with vulnerable libraries can be fixed by just updating the libraries

WHEN SOFTWARE ATTACKS!!

DEPENDENCY TYPOSQUATTING

"...into a trusted piece of software, *which is written by humans*, ..."

How this attack is executed?

- Developers can mistakenly have a typo of the package name, or think they are using a legit software component.

I would never fall for that.. right?

The Hacker News

Home | Data Breaches | Cyber Attacks | Vulnerabilities | Malware | Offers | Contact

Several Malicious Typosquatted Python Libraries Found On PyPI Repository

July 30, 2021 Ravie Lakshmanan

As many as eight Python packages that were downloaded more than 30,000 times have been removed from the PyPI portal for containing malicious code, once again highlighting how software package repositories are evolving into a popular target for supply chain attacks.

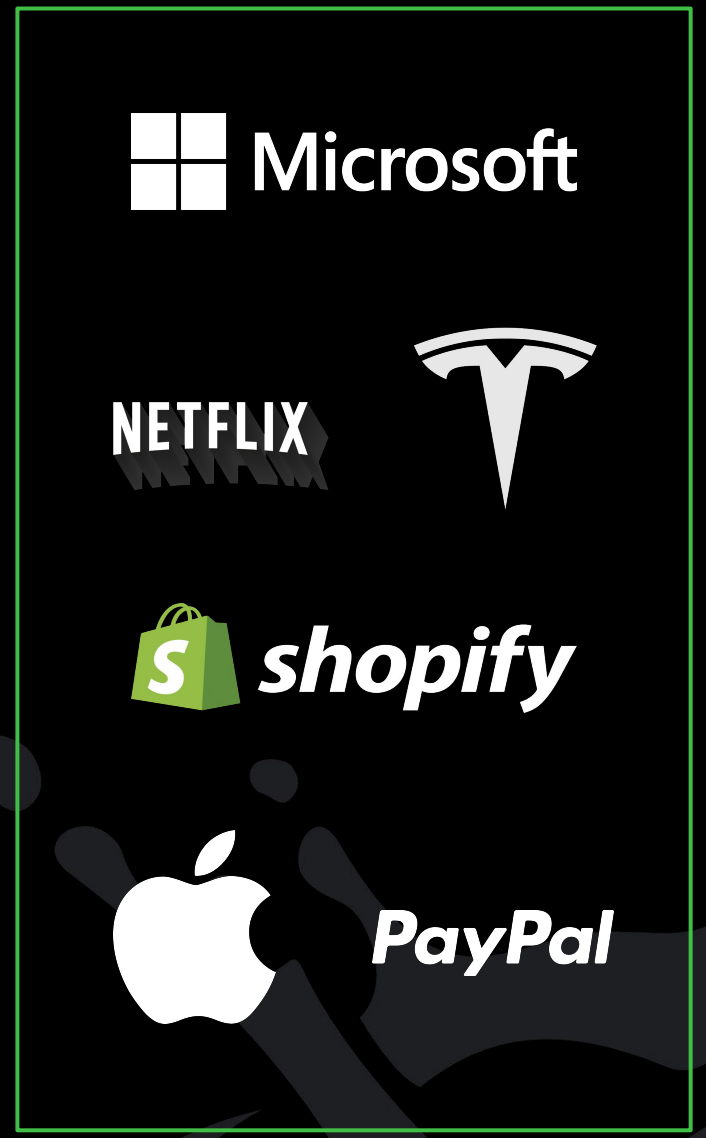
DEPENDENCY CONFUSION

“I got it, 3rd party is risky and I should keep my eyes open and monitor that. At least my company proprietary code is fine and I fully trust it!“

But...

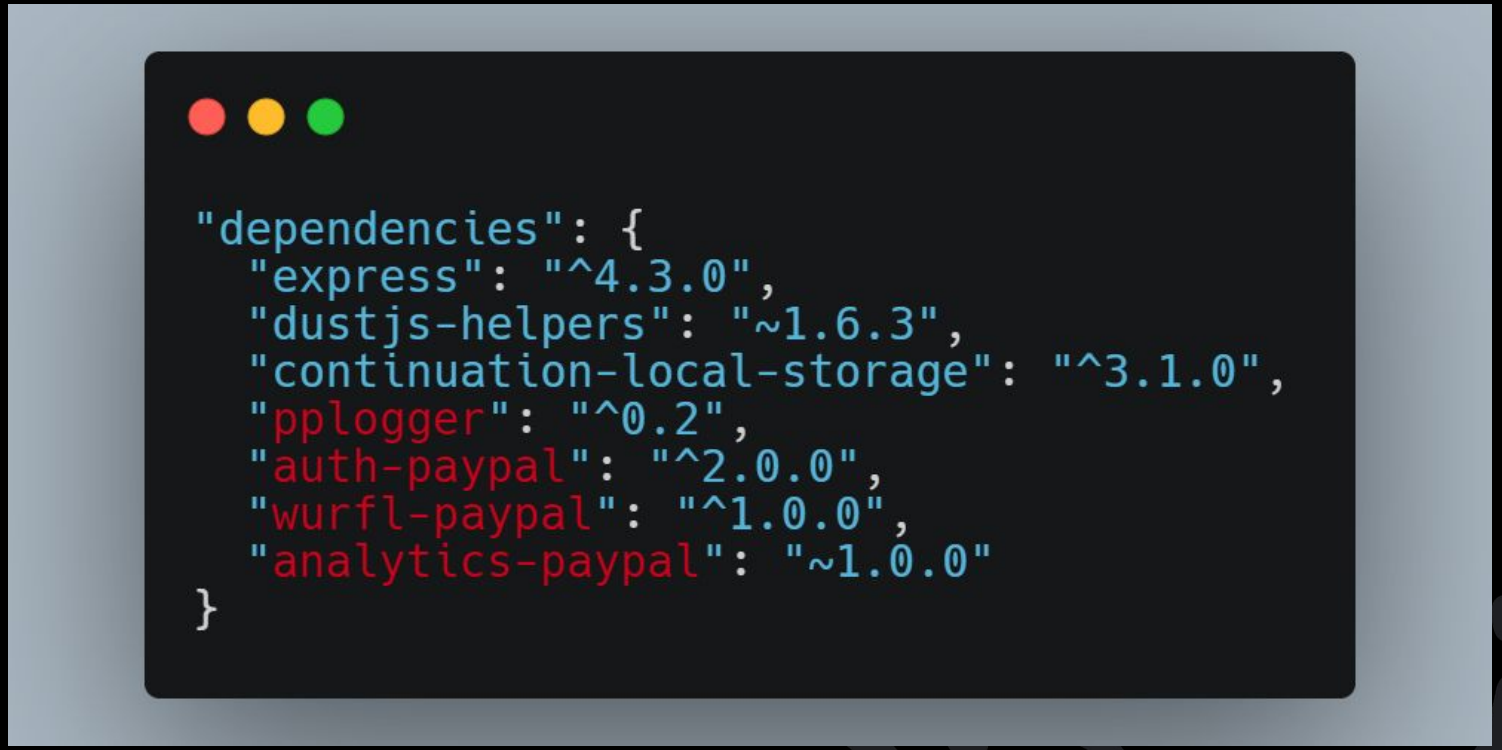
How this attack is executed?
Most build systems public repositories get priority over private repositories.

Is it real?
In the POC the security searcher was able to breach over 35 major companies' internal systems to achieve remote code execution



DEPENDENCY CONFUSION - HOW IT WORKS?

A dependency confusion attack or supply chain substitution attack occurs when a software installer script is tricked into pulling a malicious code file from a public repository instead of the intended file of the same name from an internal repository



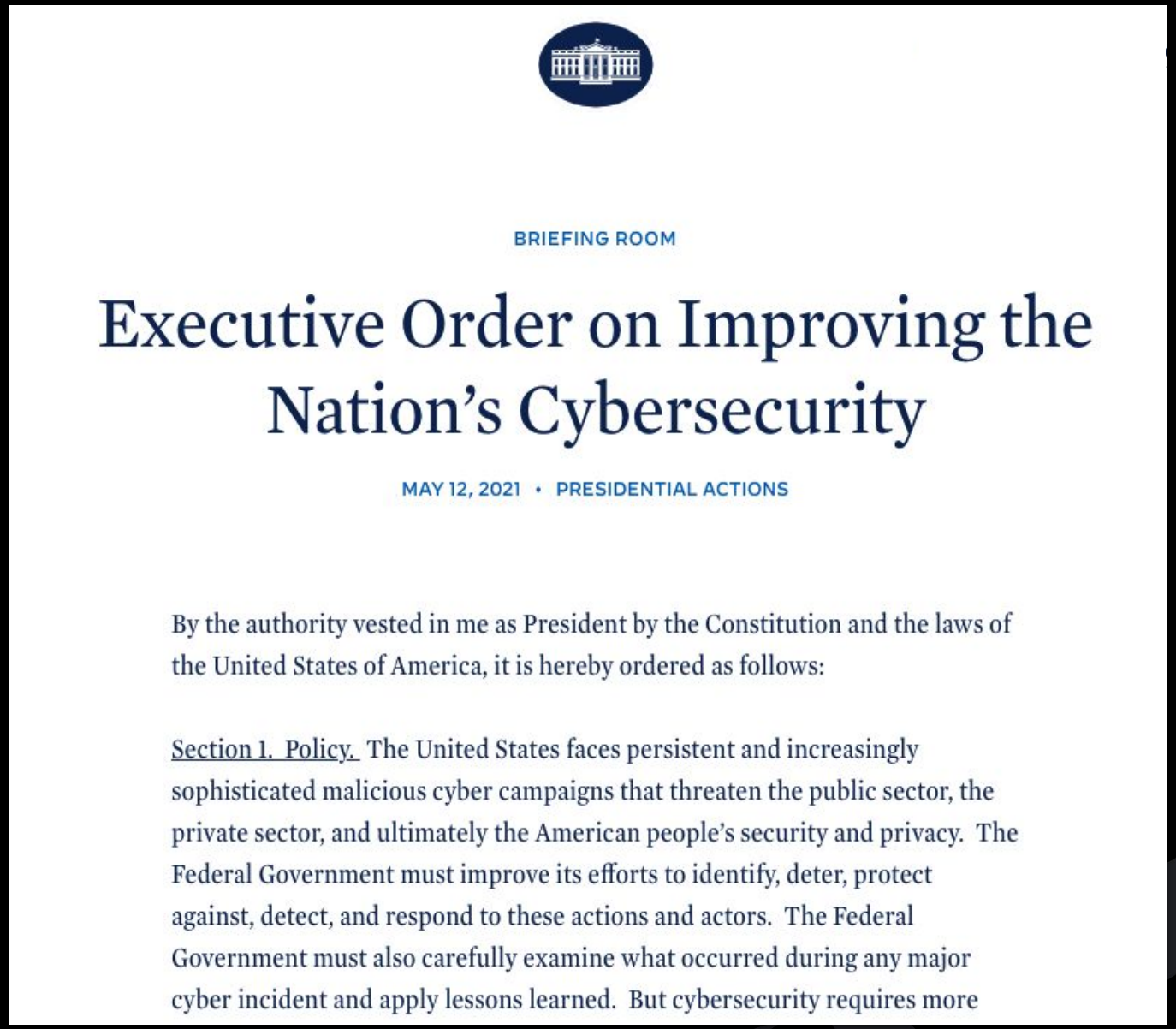
"Eighteen thousand [customers] was our best estimate of who may have downloaded the code between March and June of 2020."

-Sudhakar Ramakrishna, SolarWinds President & CEO

THE BEST DEFENSE IS A GOOD OFFENSE


JFrog
The Liquid Software Company
UNDERSTANDING WHAT MAKES UP YOUR SOFTWARE

THE UNITED STATES GOVERNMENT EQUATES CYBERSECURITY WITH NATIONAL SECURITY.



BRIEFING ROOM

Executive Order on Improving the Nation's Cybersecurity

MAY 12, 2021 • PRESIDENTIAL ACTIONS

By the authority vested in me as President by the Constitution and the laws of the United States of America, it is hereby ordered as follows:

Section 1. Policy. The United States faces persistent and increasingly sophisticated malicious cyber campaigns that threaten the public sector, the private sector, and ultimately the American people's security and privacy. The Federal Government must improve its efforts to identify, deter, protect against, detect, and respond to these actions and actors. The Federal Government must also carefully examine what occurred during any major cyber incident and apply lessons learned. But cybersecurity requires more

Sec. 4. Enhancing Software Supply Chain Security

(vii) providing a purchaser a Software Bill of Materials (SBOM) for each product directly or by publishing it on a public website;

(j) the term "Software Bill of Materials" or "SBOM" means a formal record containing the details and supply chain relationships of various components used in building software. Software developers and vendors often create products by assembling existing open source and commercial software components. The SBOM enumerates these components in a product. It is analogous to a list of ingredients on food packaging. An SBOM is useful to those who develop or manufacture software, those who select or purchase software, and those who operate software. Developers often use available open source and third-party software components to create a product; an SBOM allows the builder to make sure those components are up to date and to respond quickly to new vulnerabilities. Buyers can use an SBOM to perform vulnerability or license analysis, both of which can be used to evaluate risk in a product. Those who operate software can use SBOMs to quickly and easily determine whether they are at potential risk of a newly discovered vulnerability. A widely used, machine-readable SBOM format allows for greater benefits through automation and tool integration. The SBOMs gain greater value when collectively stored in a repository that can be easily queried by other applications and systems. Understanding the supply chain of

Nutrition Facts

Serving Size 1/12 pkg (45g mix)
Servings Per Container 12

Amount Per Serving	Mix	Prepared
Calories	170	270
Calories from Fat	10	100
	% Daily Value**	
Total Fat 1.5g*	2%	17%
Saturated Fat 0g	0%	29%
Trans Fat 0g		
Cholesterol 0mg	0%	25%
Sodium 270mg	11%	15%
Potassium 95mg	3%	5%
Total Carbohydrate 38g	13%	13%
Dietary Fiber less than 1g	4%	4%
Sugars 21g		
Protein 2g		
Vitamin A	0%	8%
Calcium	2%	6%
Iron	8%	10%

Not a significant source of vitamin C.

* Amount in mix. As prepared, one serving provides 11g total fat (6g saturated fat), 75mg cholesterol, 350mg sodium, 160mg potassium, 40g total carbohydrate (23g sugars), and 5g protein.

** Percent Daily Values are based on a 2,000 calorie diet. Your daily values may be higher or lower depending on your calorie needs:

	Calories	2,000	2,500
Total Fat	Less than	65g	80g
Sat Fat	Less than	20g	25g
Cholesterol	Less than	300mg	300mg
Sodium	Less than	2,400mg	2,400mg
Potassium		3,500mg	3,500mg
Total Carbohydrate		300g	375g
Dietary Fiber		25g	30g

Ingredients: Sugar, Enriched Flour Bleached (wheat flour, niacin, iron, thiamin mononitrate, riboflavin, folic acid), **Cocoa Processed with Alkali, Corn Starch, Canola Oil, Baking Powder** (baking soda, sodium aluminum sulfate, monocalcium phosphate), **Salt.**

CONTAINS WHEAT; MAY CONTAIN MILK INGREDIENTS.

DISTRIBUTED BY GEN… …LS SALES, INC., MINNEAPOLIS… …440 USA

© General Mil… …40775102

Partially Produced wi… …hetic Engineering

Learn more at Ask.GeneralMil…

Ingredients

THE NATIONAL TELECOMMUNICATIONS AND INFORMATION ADMINISTRATION

Started discussion in 2018

The Software Transparency Project

For Medical Device Manufacturers

A common method for safety guidelines with developing software

Inform Purchasers of devices of the software

WHAT IS A SOFTWARE BILL OF MATERIALS?

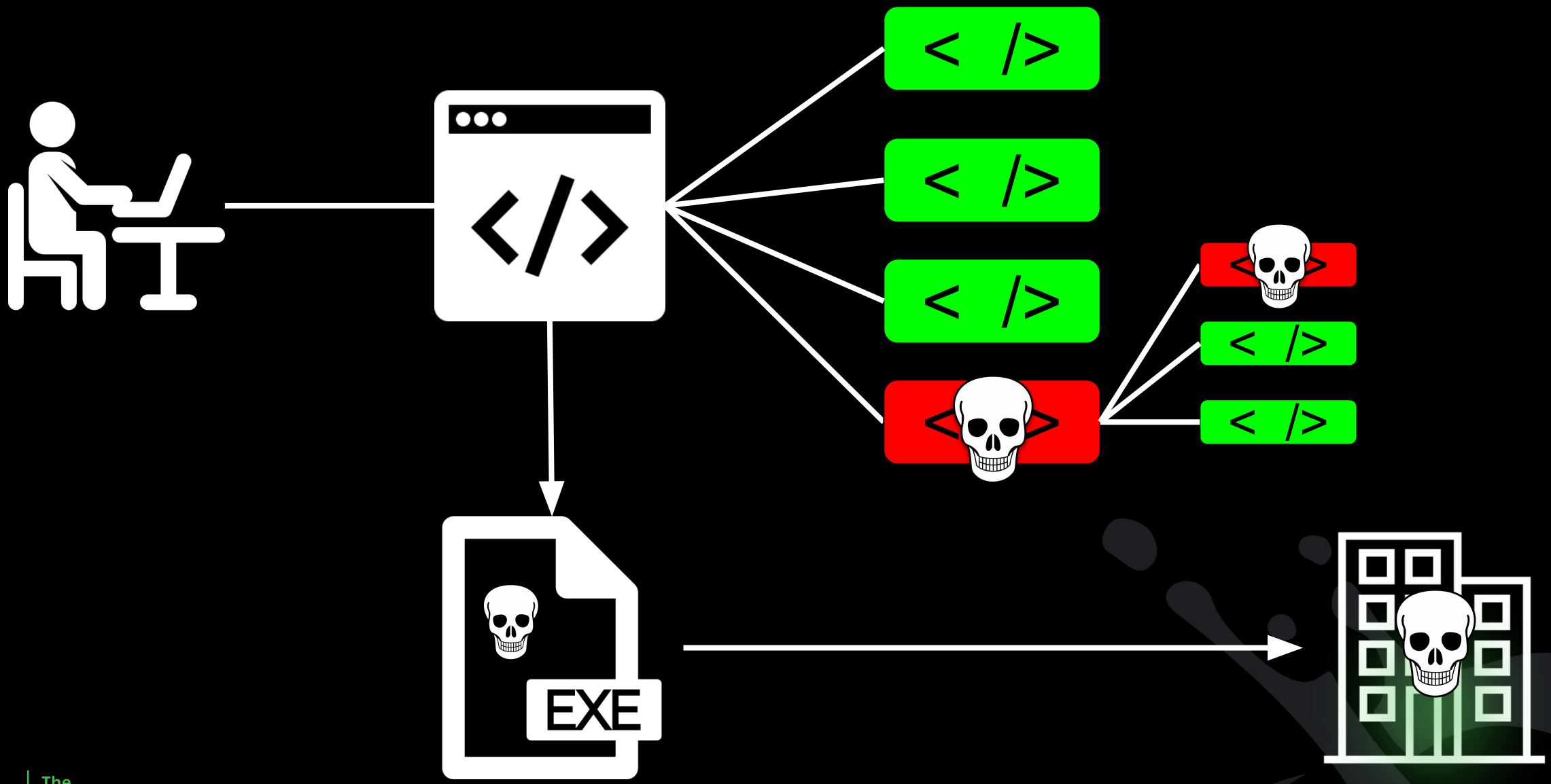
- A list of ingredients that makes up what's inside of software
- Including libraries and modules, can be open source or proprietary, free or paid, and the data can be widely available or access-restricted.
- Additionally, tooling, environmental information, settings, versions, etc.

WHO USES SBOM AND FOR WHAT?

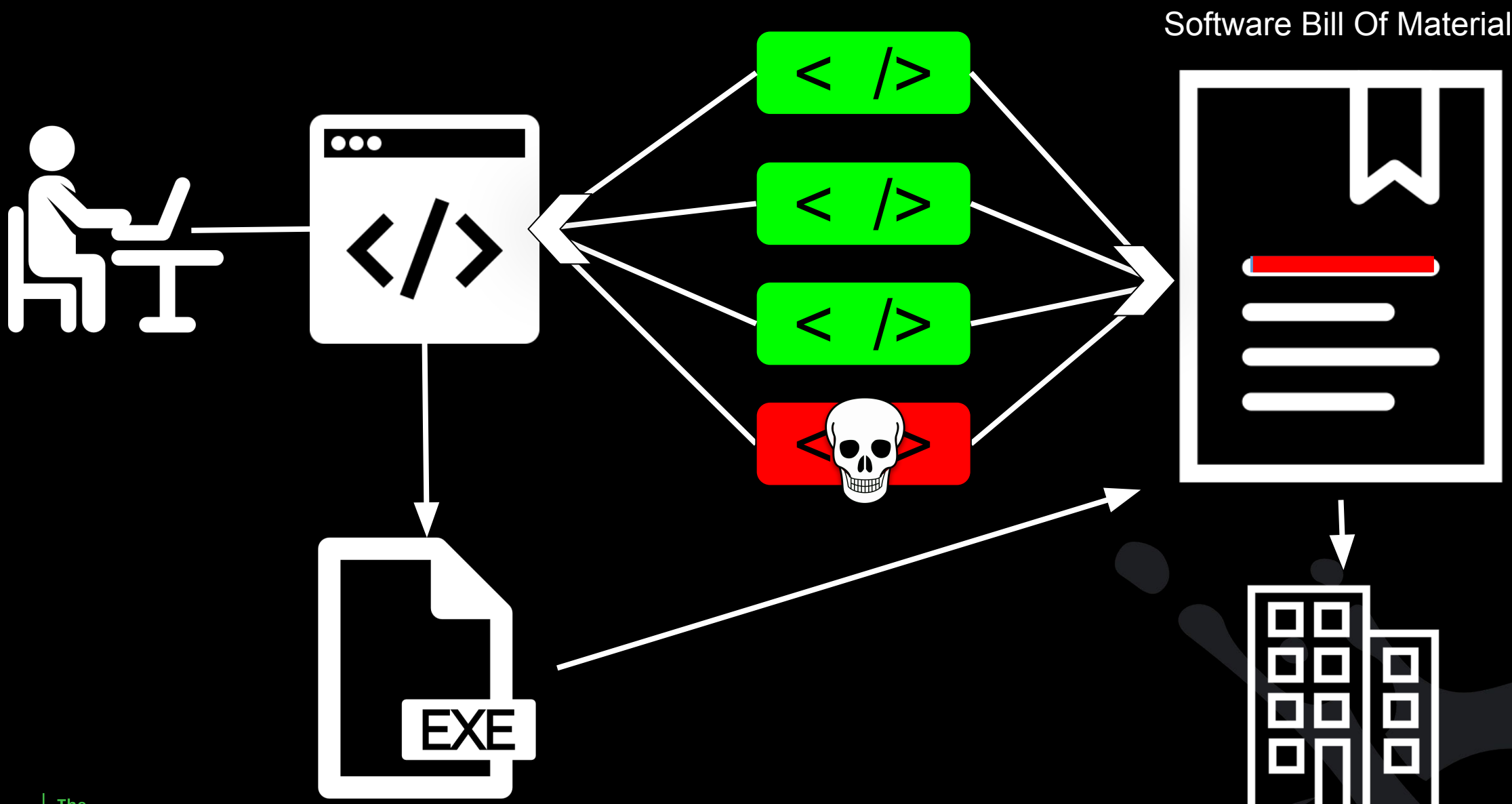
- For those who produce software, SBOMs are used to assist in the building and maintenance of their software, including upstream components.
- For those who choose or purchase software, SBOMs are used to inform pre-purchase assurance, negotiate discounts, or plan implementation strategies.
- For those who operate software, SBOMs are used to inform vulnerability management and asset management, to manage licensing and compliance, and to quickly identify software or component dependencies and supply chain risks.

WHAT ARE THE BENEFITS OF SBOM?

- Identifying, mitigate, and avoiding known vulnerabilities(including patching and compensating controls for new vulnerabilities)
- Quantifying and managing licenses
- Identifying both security and license compliance requirements
- Enabling quantification of the risks inherent in a software package
- Comprehensive information on what environment and what setting were used
- Lower operating costs due to improved efficiencies and reduced unplanned and unscheduled work.



JFrog
The Liquid Software Company



Software Releases

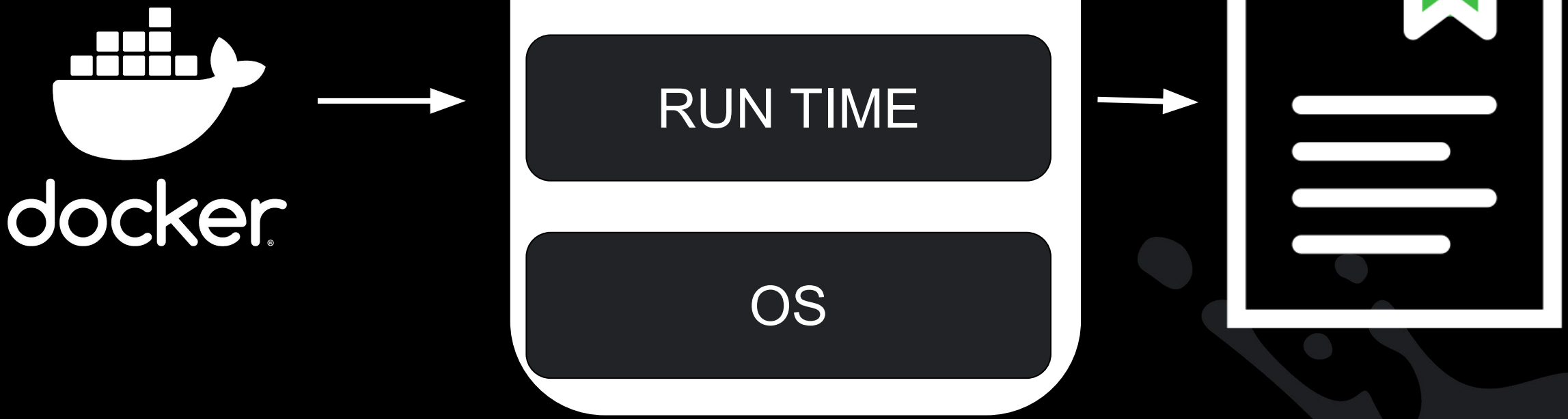
Software Bill Of Material

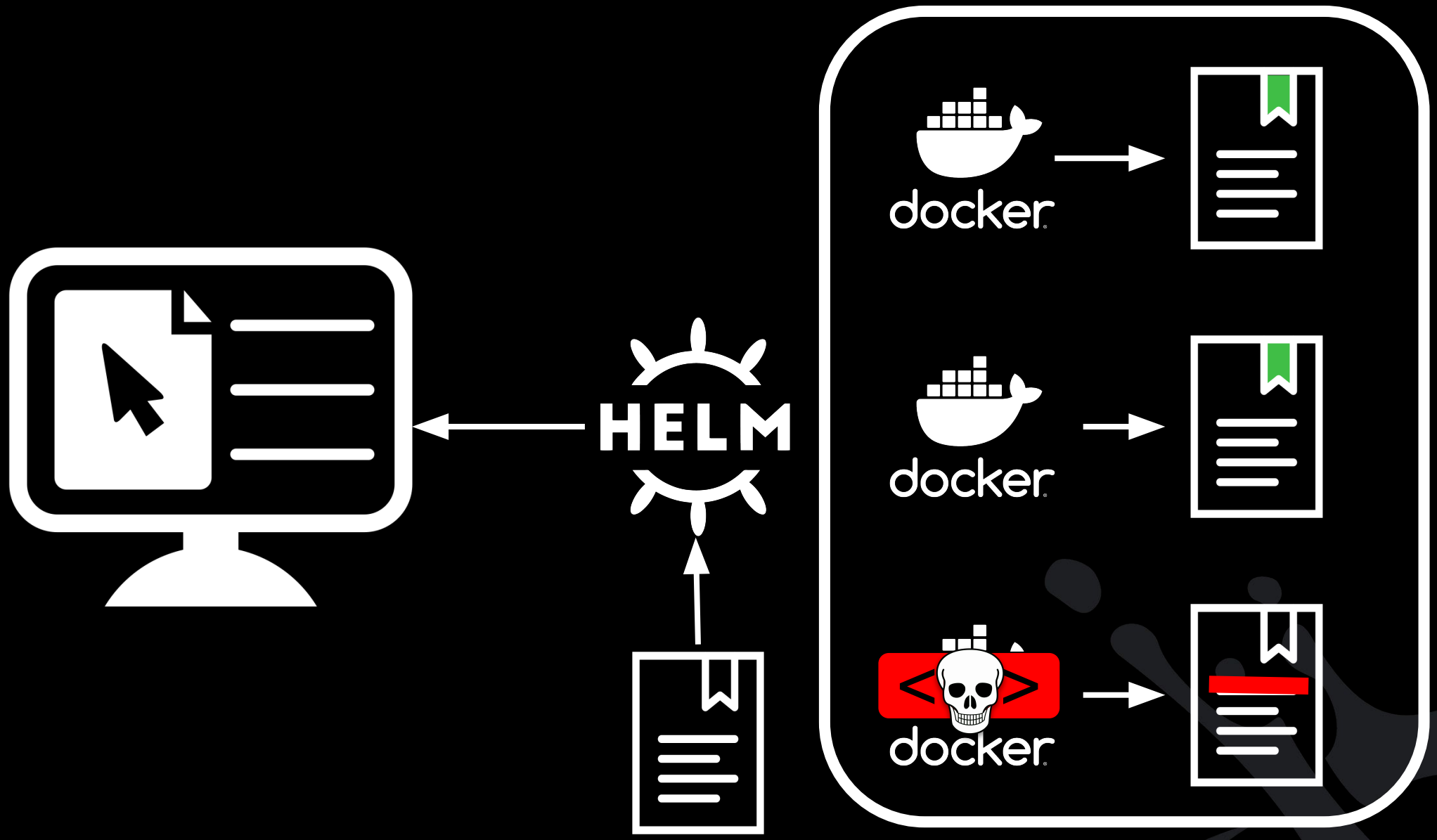
Version 1.1

Version 1.2

Version 1.3

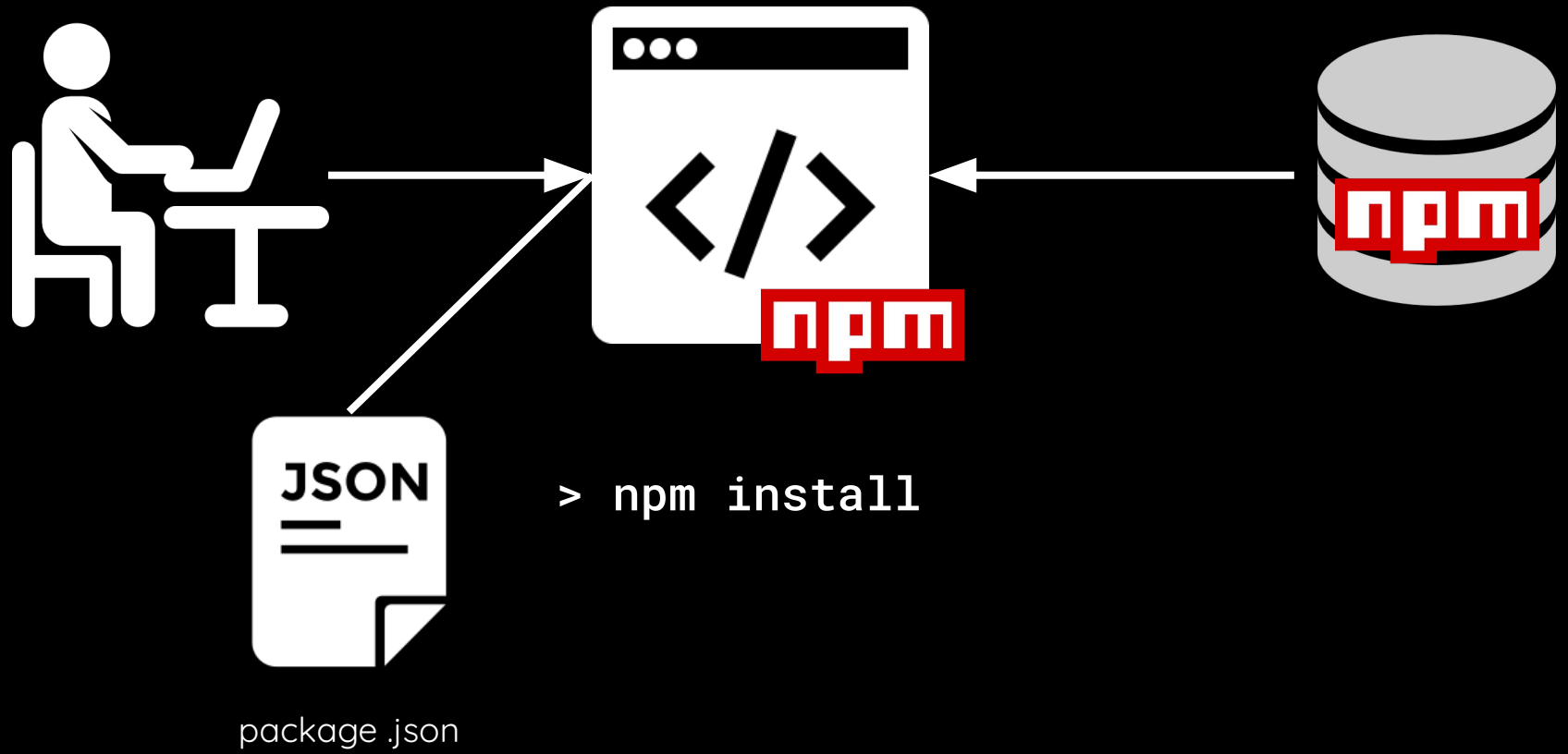
Version 1.4



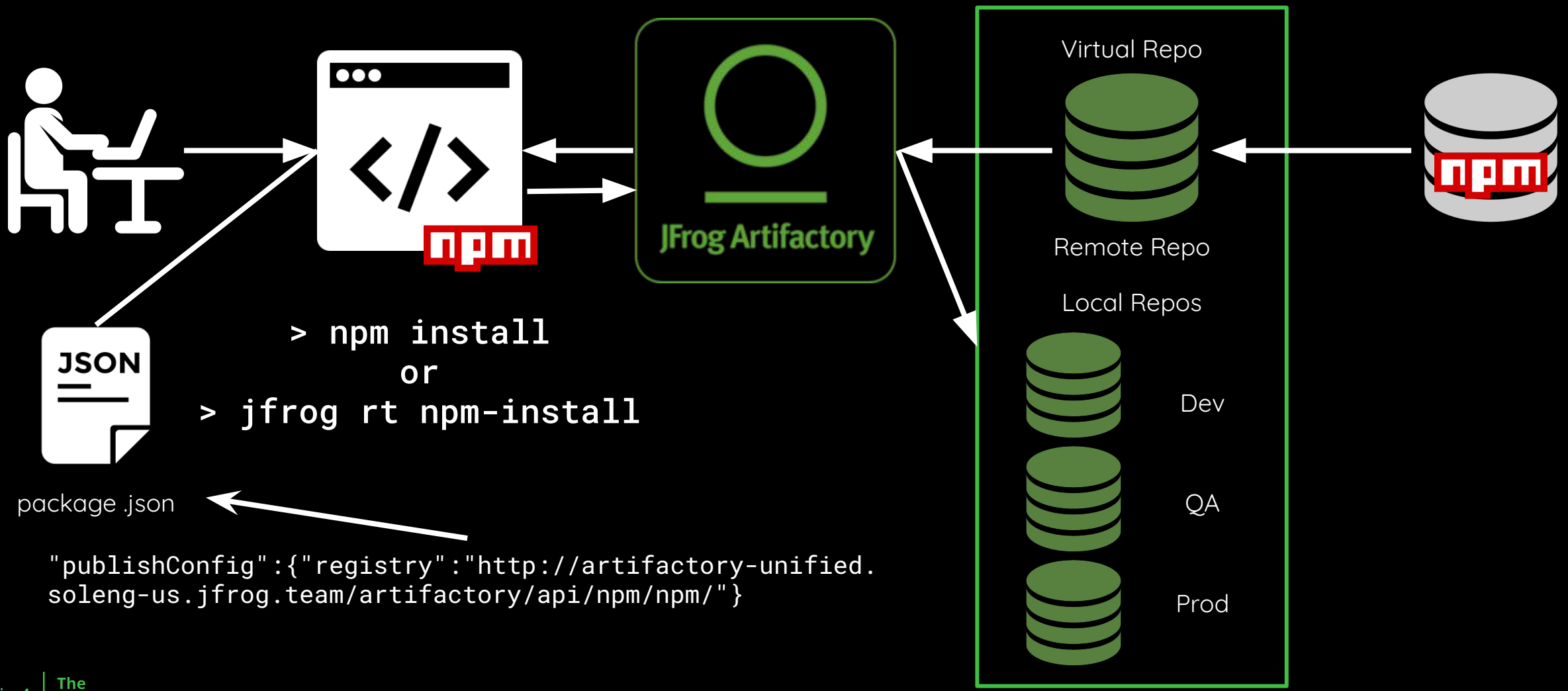



JFrog
The Liquid Software Company

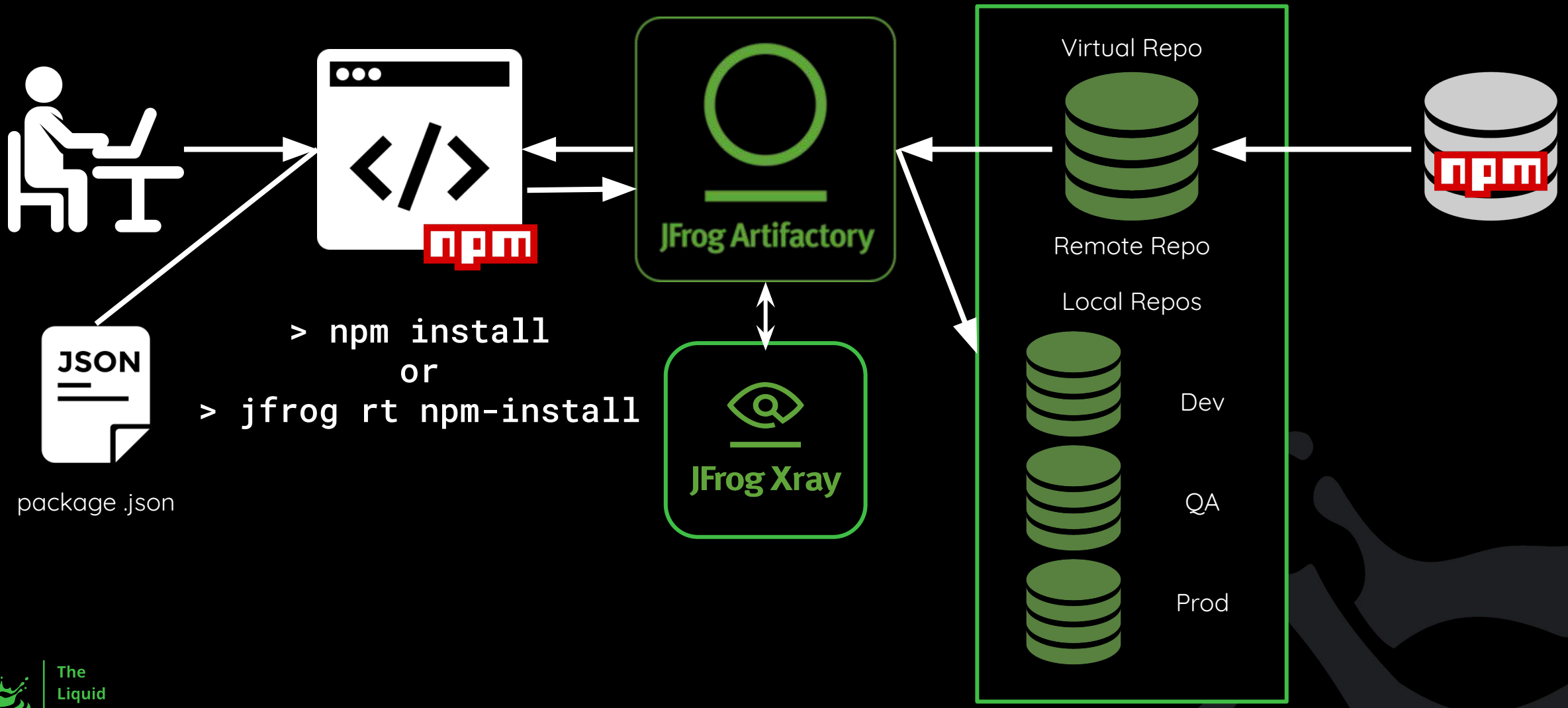
TYPICAL DEVELOPMENT WORKFLOW



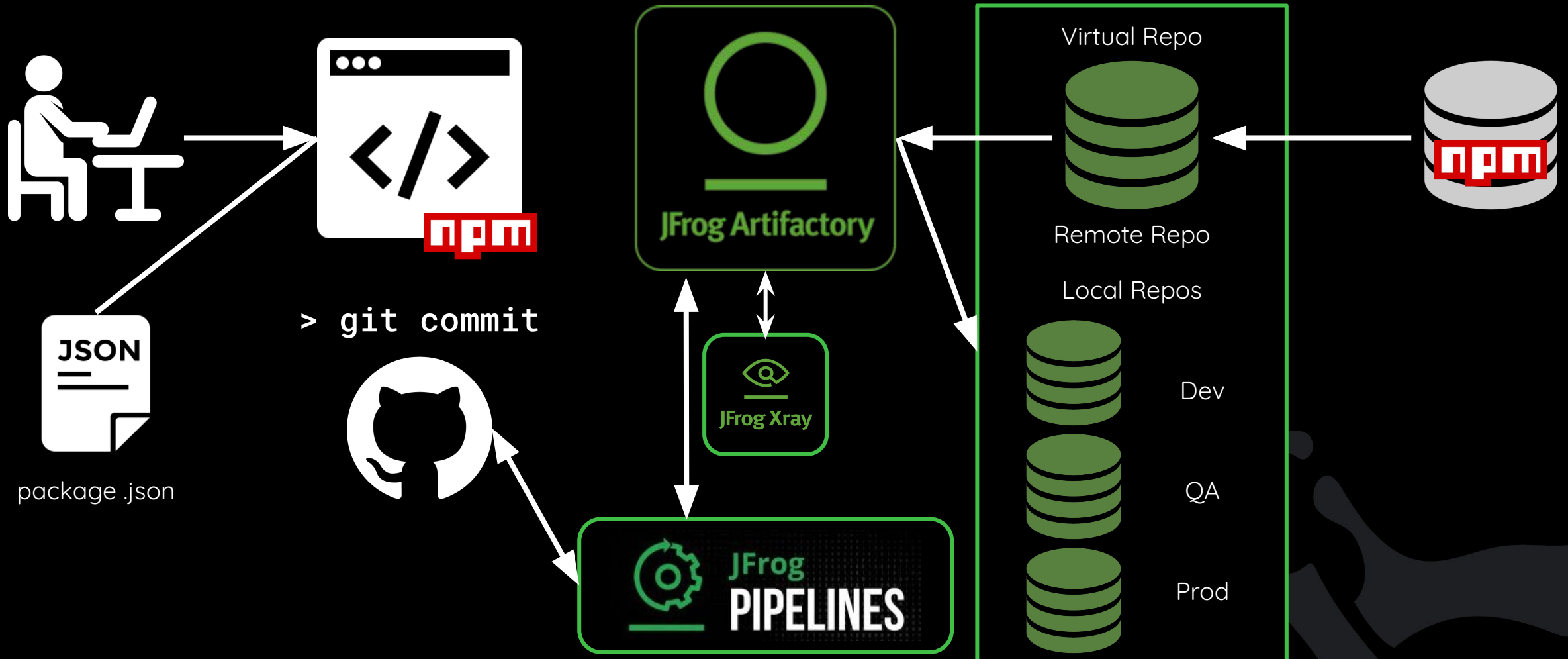
LET'S ADD ARTIFACTORY INTO YOUR PROCESS



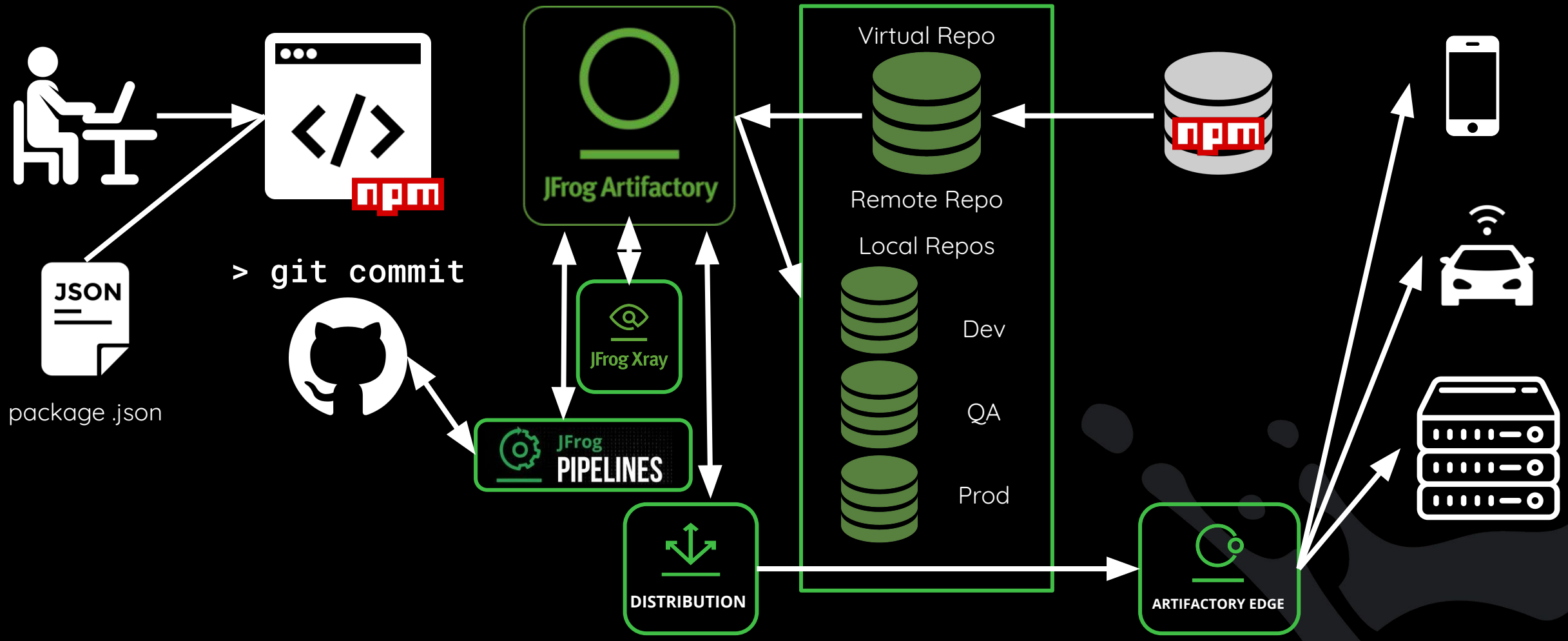
LET'S ADD SOME SECURITY AND COMPLIANCE



HOW DO WE AUTOMATE OUR PROJECT?



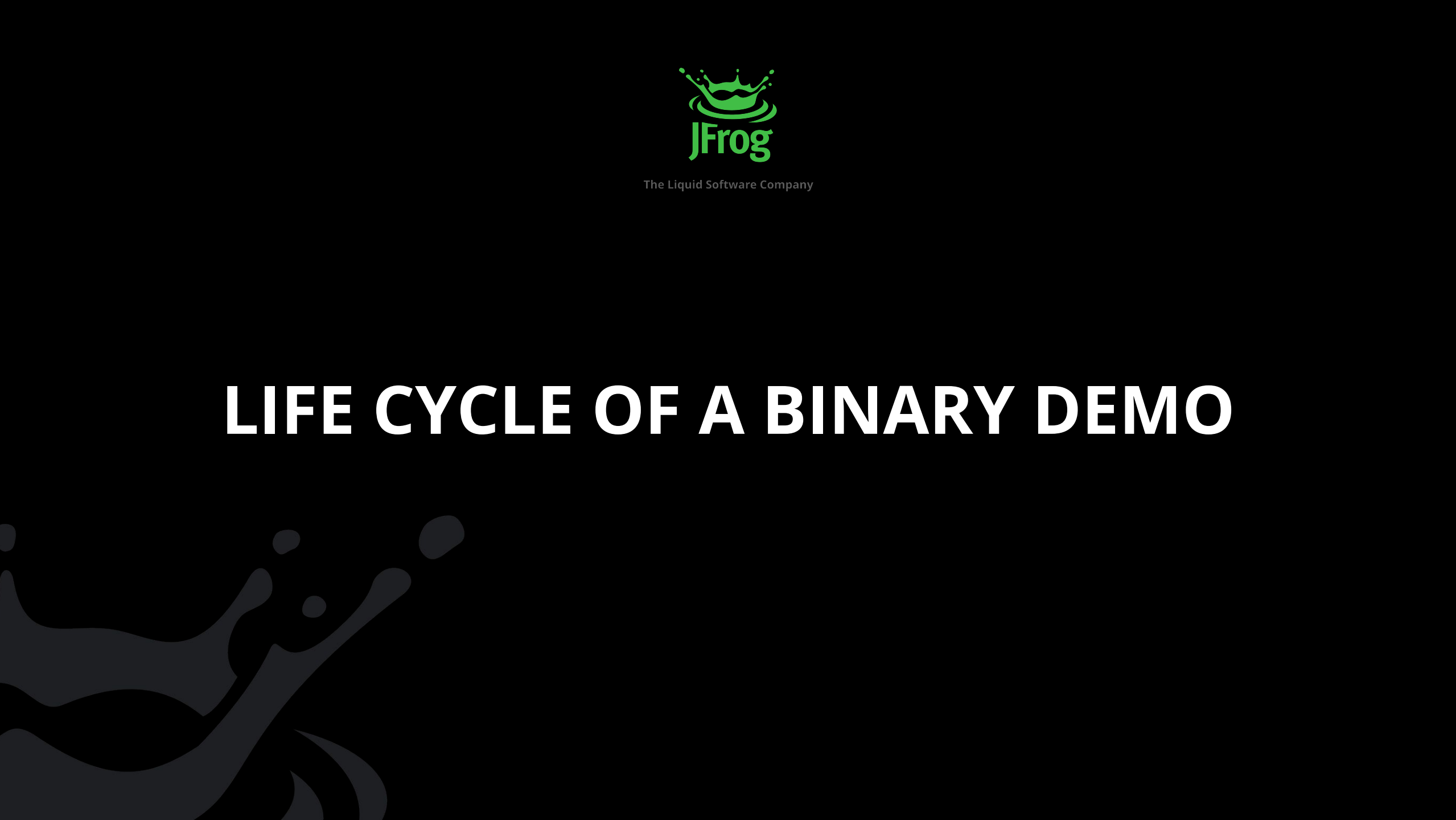
RELEASE MANAGEMENT



JFrog

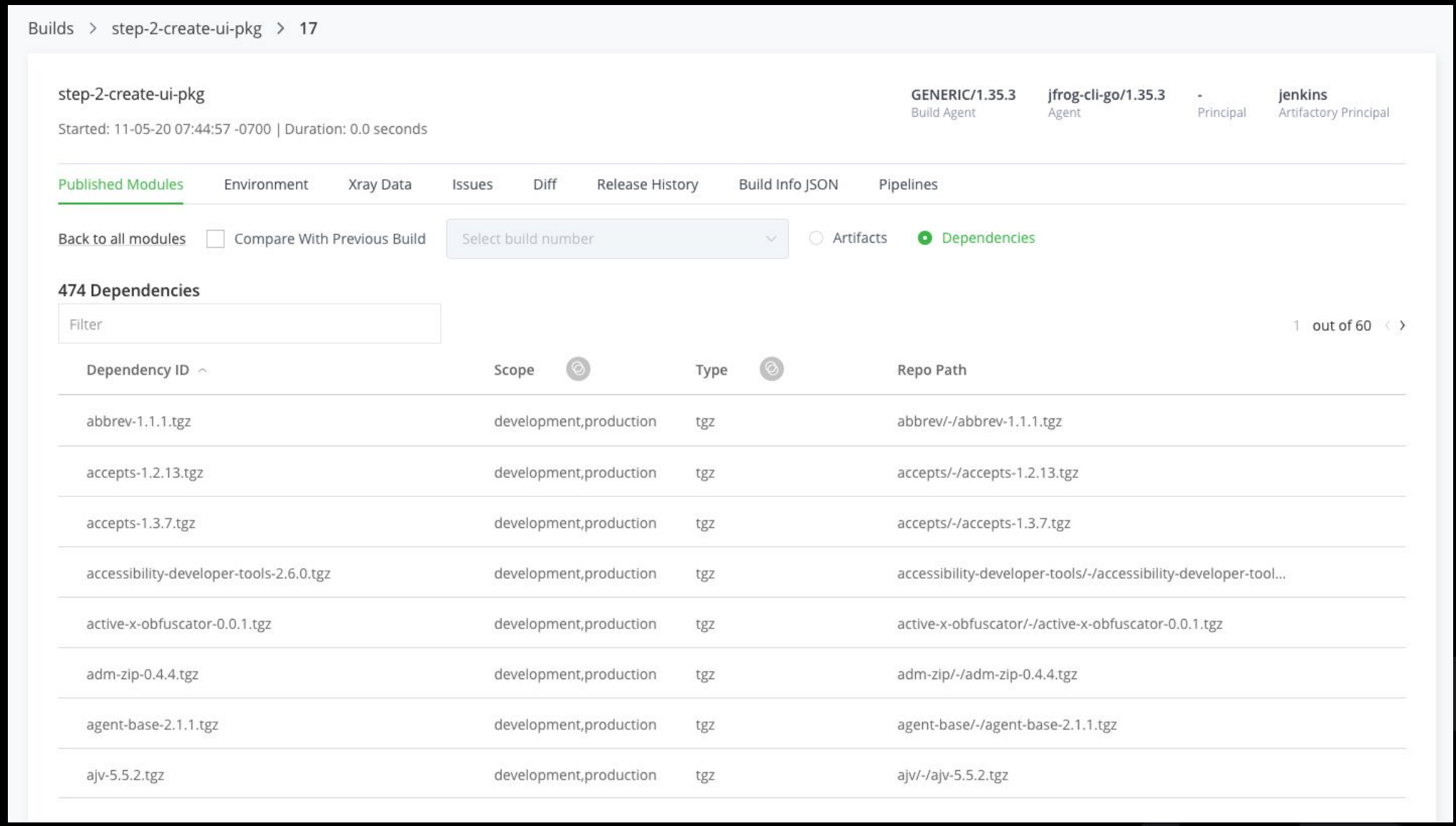
The Liquid Software Company

BREAK

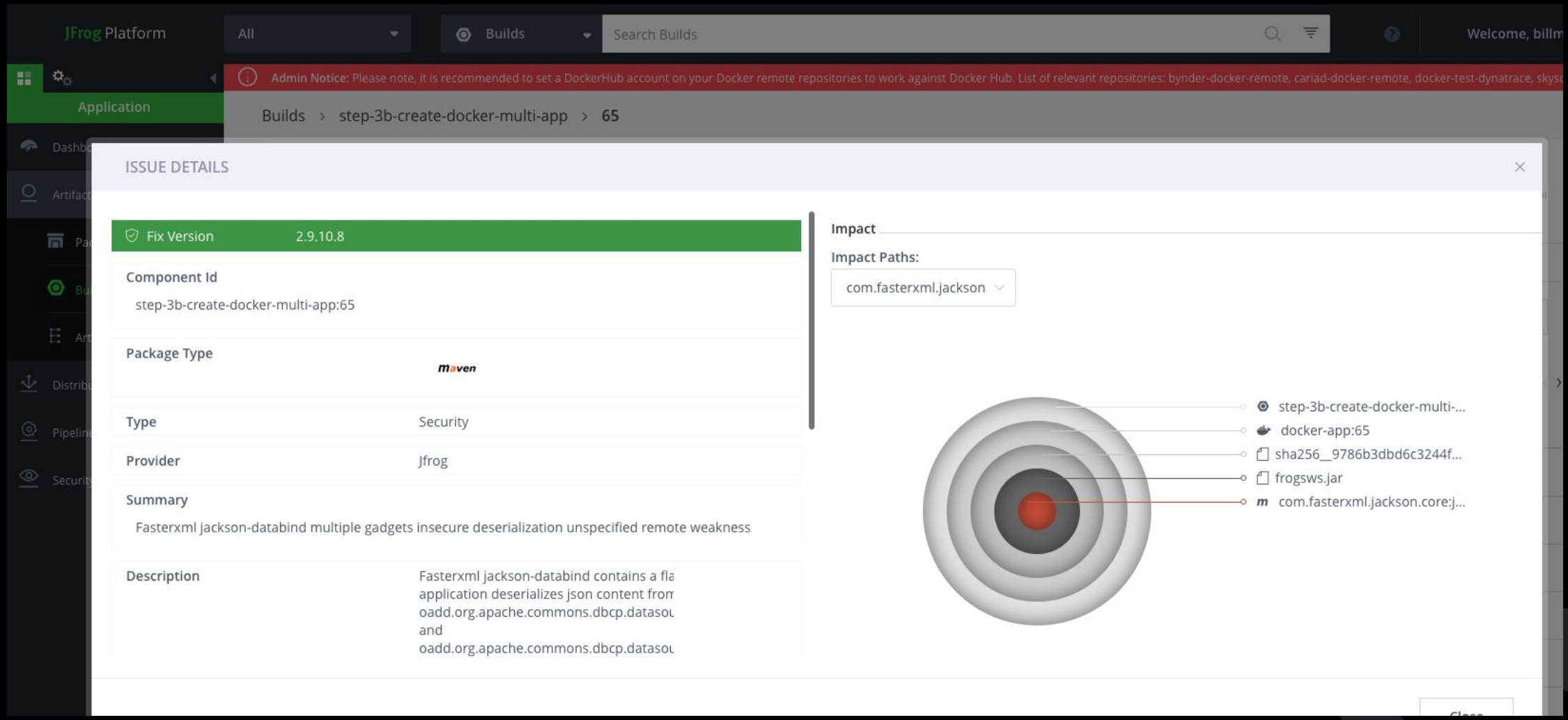


LIFE CYCLE OF A BINARY DEMO

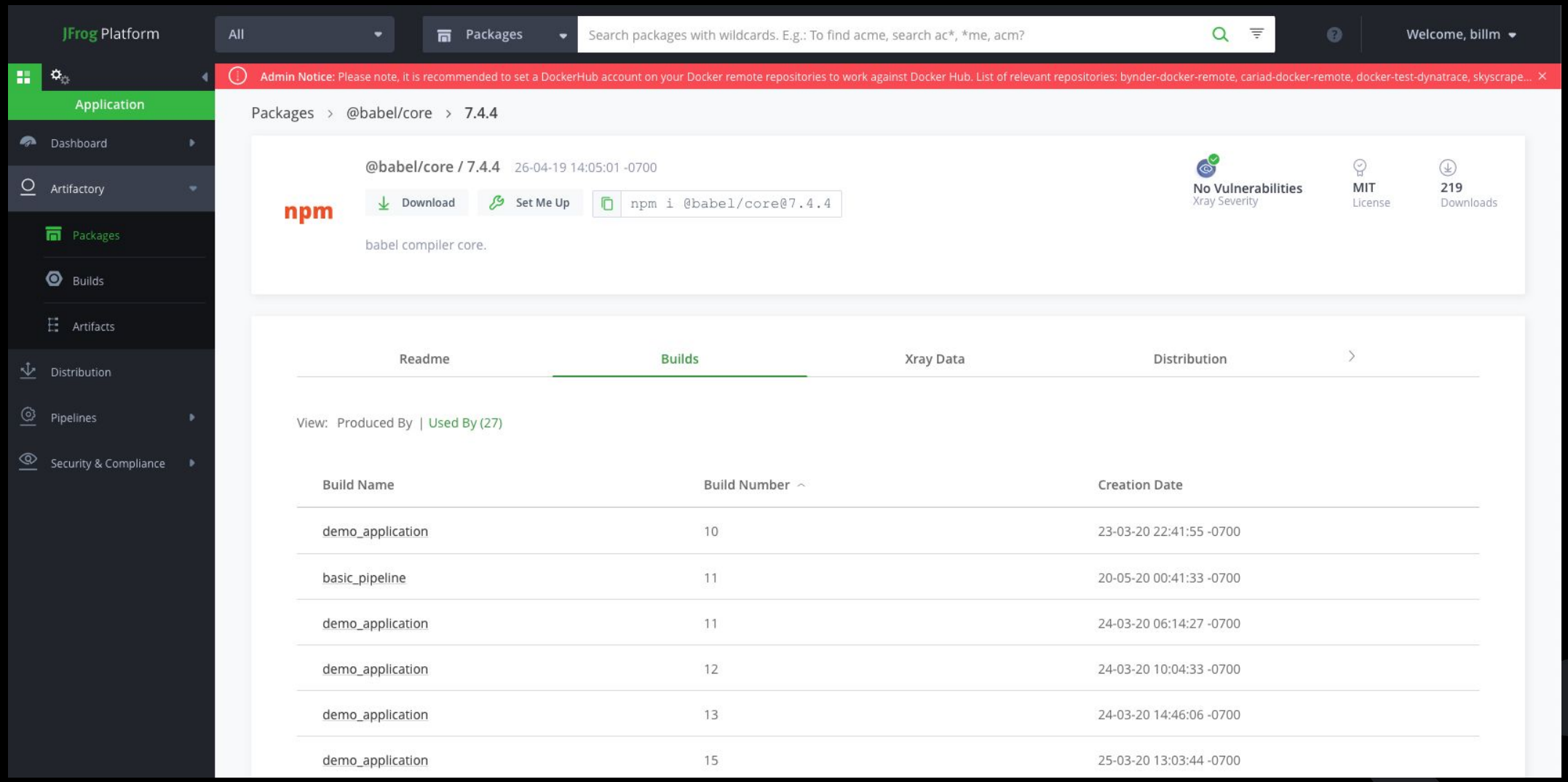
ARTIFACTORY BUILDINFO



HOW DEEP DO VULNERABILITIES GO?

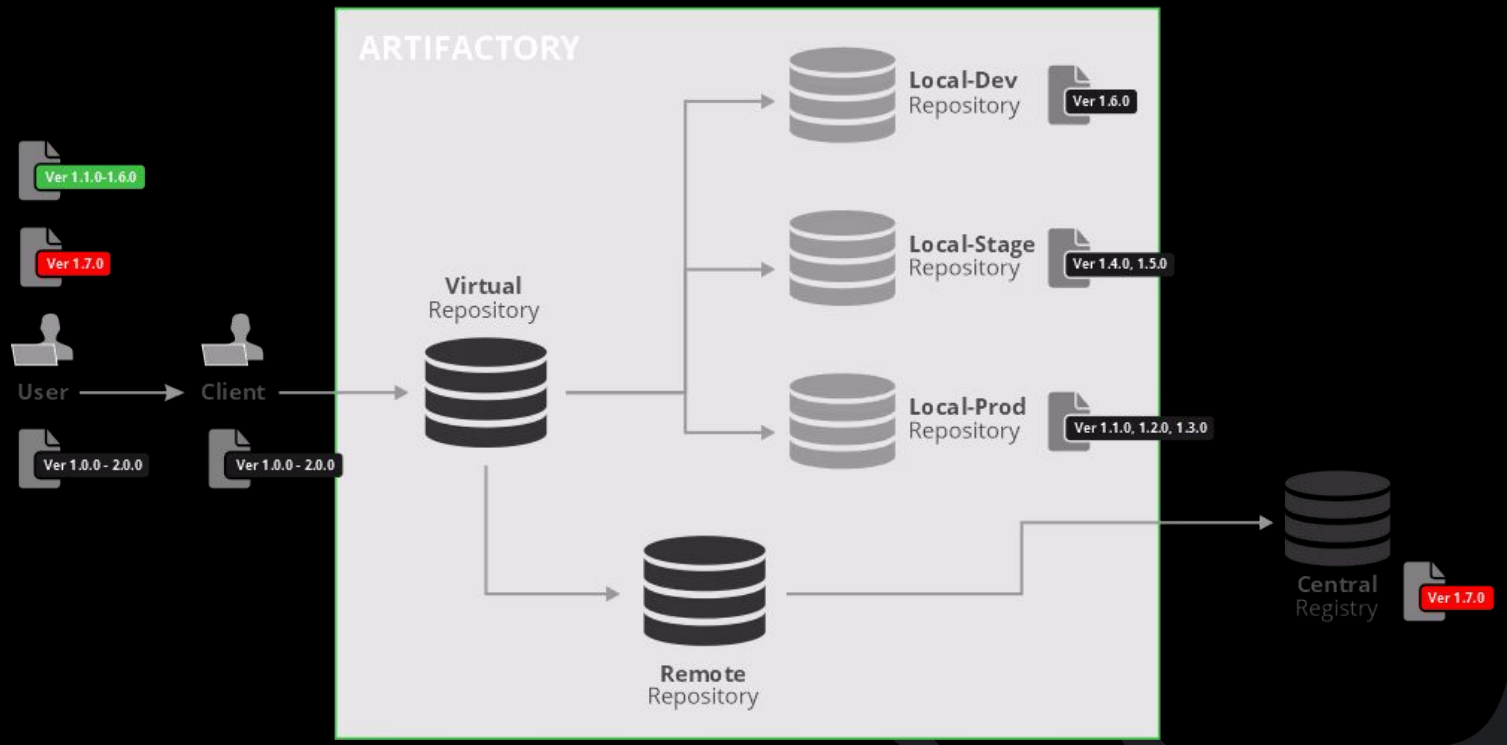


TRACING BLAST RADIUS OF BINARIES

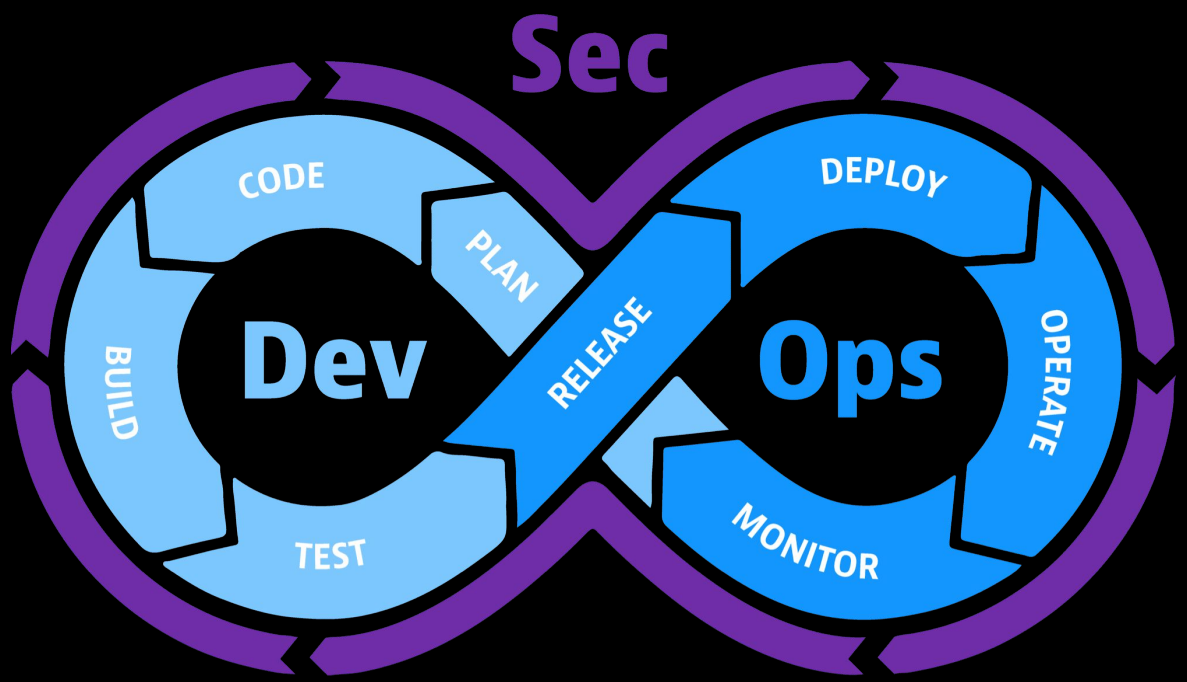


REPOSITORY PRIORITIZATION

- Set a local or Remote repositories as “Safe” by enabling “Priority Resolution”
- Resolution order precedence when searching for artifacts in virtual repositories
- Provides better resolution of Looking up Resources(repos, builds)



SBOM - ADDITIONAL INFORMATION



What Continuous Integration (CI) Tool?

When was the software built?

What stages of the Software Development Life Cycle did it go though?

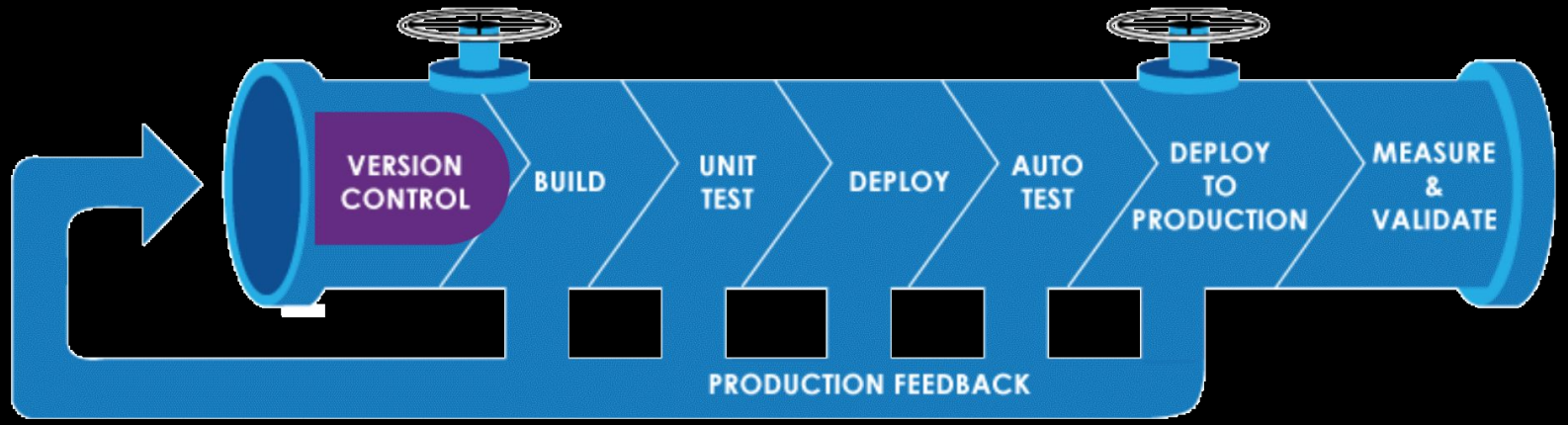
What FOSS was used in the process?

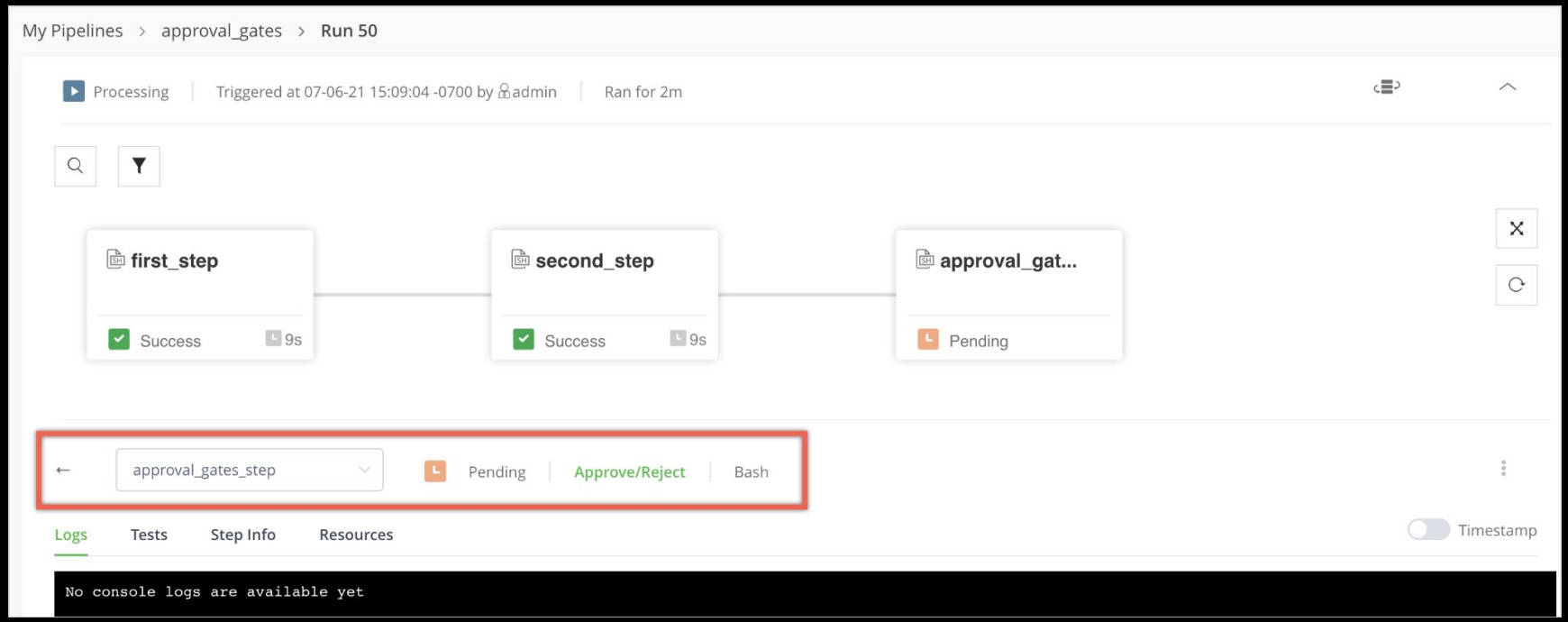
What was Environment?

What were the settings in that Environment?

Were they any Security or Vulnerabilities?

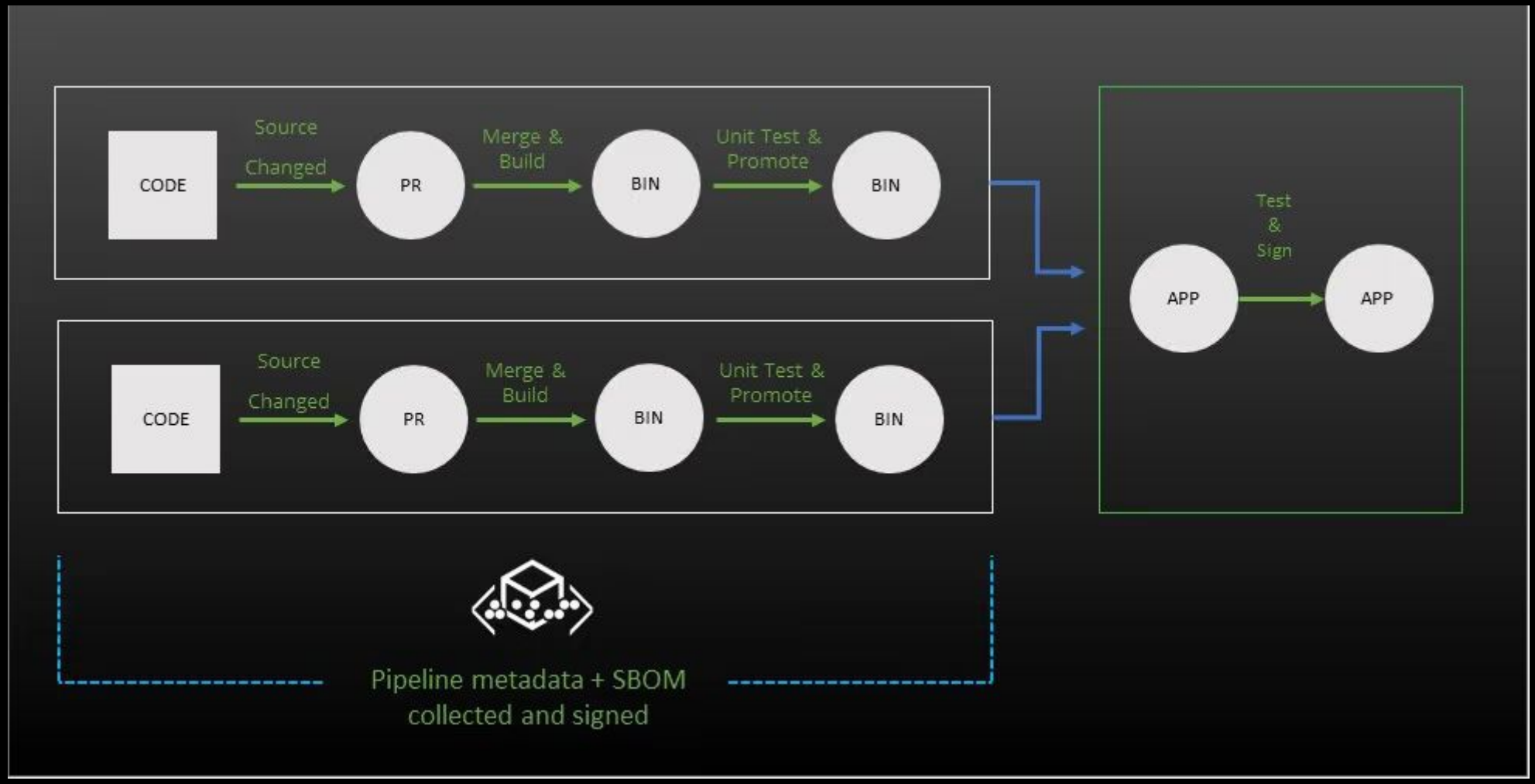
PROMOTION AND APPROVALS





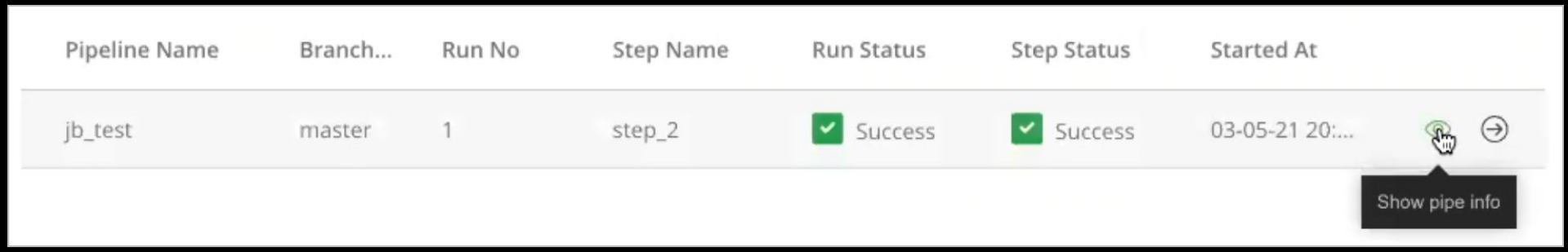
The Liquid Software Company

SIGNED PIPELINES



SIGNED PIPELINES

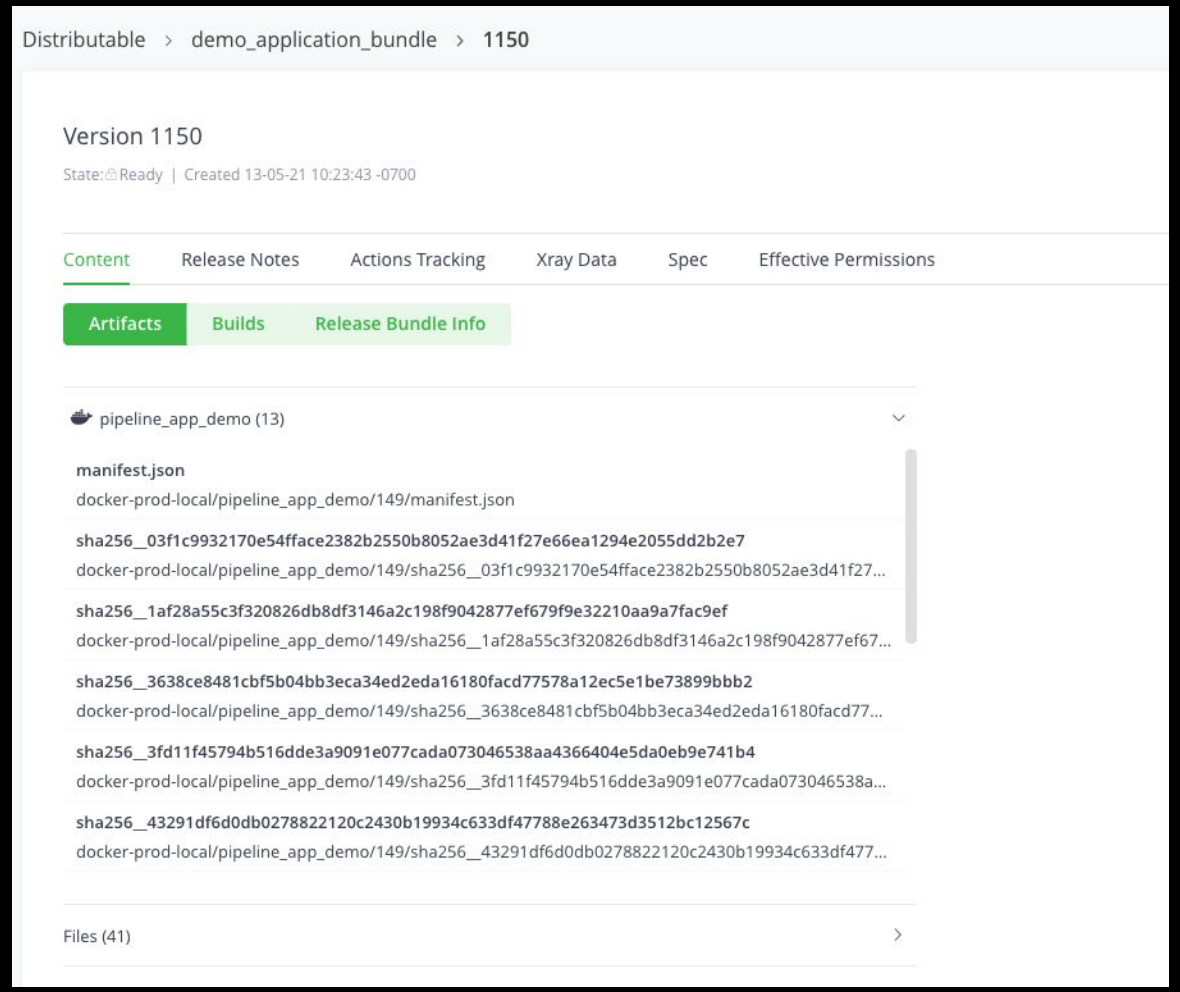
- Signed Pipelines builds comprehensive metadata, called **pipeinfo**



- Provides complete visibility and audit for each step and run, which can be viewed in the UI.
- The promotion of the builds, release bundles, or deployments can be blocked if authenticity can not be verified

JFROG RELEASE BUNDLE

- Immutable Bill of Materials (BOM)
- Verified distribution: Secure data in transit with verification at consumption point
- Fine-grained Role-Based Access Control (RBAC) - for publishing, managing and consuming binaries, across all internal/external targets
- JFrog Xray security data can block distribution
- JFrog Pipelines has native steps for Distribution



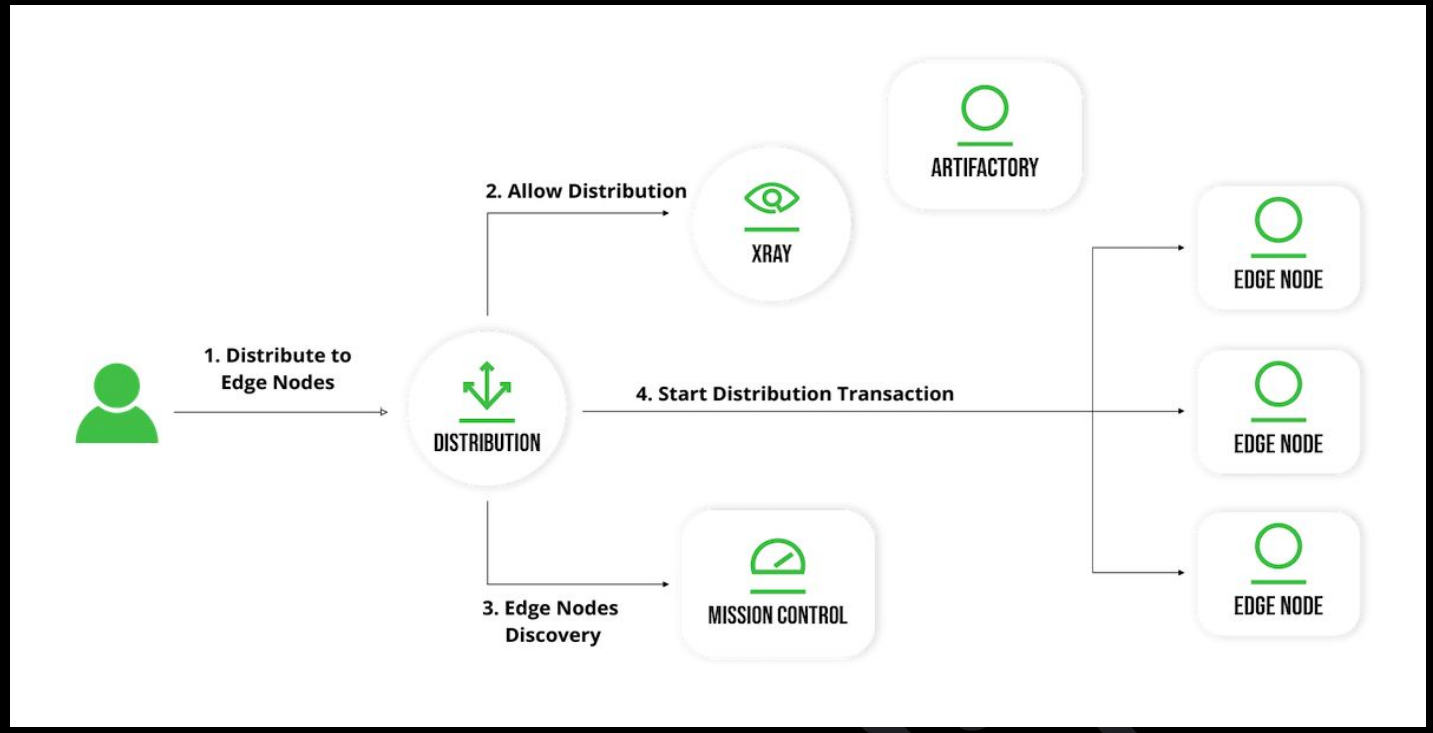
JFROG DISTRIBUTION & TRACKING

Where and when something was deployed

Extending the SBOM to the implementation

Ensure the integrity of the software

Update the SBOM at the last mile



The Liquid Software Company

SBOM MISCONCEPTIONS

- Won’t SBOMs be a “roadmap to the attacker”?
- Does an SBOM require source code disclosure?
- Does a list of the software components I include expose my intellectual property?

SBOM - Software Bill of Materials

Needed to work with the US Government

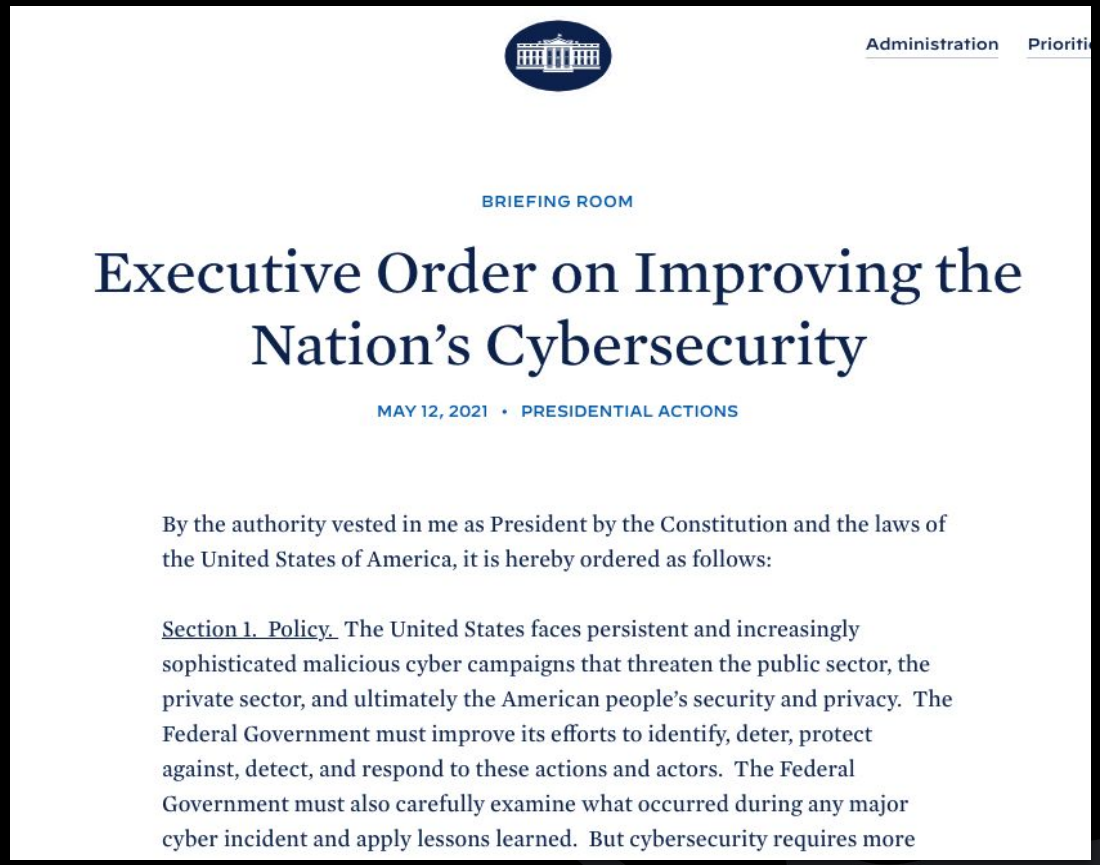
A complete list of all FOSS Libraries used in the software produced

How, What, and When was it made?

Ability to Audit and Trace anything that is potentially threatening

Complete accountability for all software produced

Security and License Compliance



Administration Prioriti

BRIEFING ROOM

Executive Order on Improving the Nation's Cybersecurity

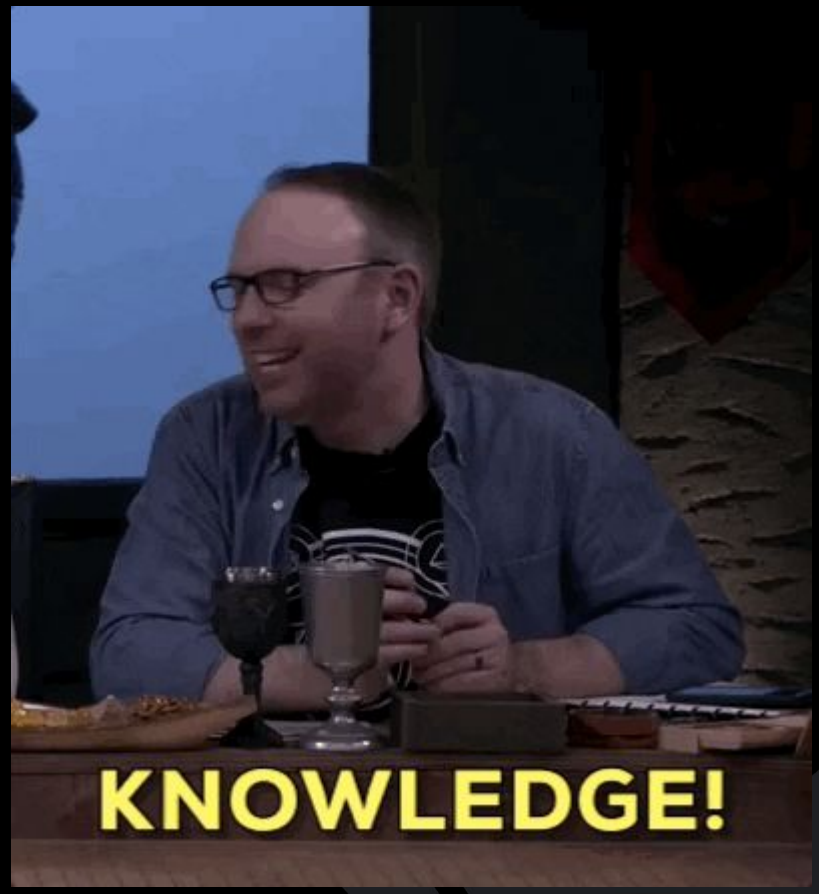
MAY 12, 2021 • PRESIDENTIAL ACTIONS

By the authority vested in me as President by the Constitution and the laws of the United States of America, it is hereby ordered as follows:

Section 1. Policy. The United States faces persistent and increasingly sophisticated malicious cyber campaigns that threaten the public sector, the private sector, and ultimately the American people's security and privacy. The Federal Government must improve its efforts to identify, deter, protect against, detect, and respond to these actions and actors. The Federal Government must also carefully examine what occurred during any major cyber incident and apply lessons learned. But cybersecurity requires more

We learned something ....

- ✓ You have the power to protect yourself and your company!!
- ✓ Not all hope is lost!! It is a winnable battle
- ✓ How to protect yourself against typosquatting attack
- ✓ How to protect yourself against dependency confusion attack
- ✓ How to generate and manage SBOM for traceability and audit
- ✓ Best practices on protecting your binary during distribution
- ✓ Best practices on securing your pipelines



The Liquid Software Company

Thank you

Dependency Typosquatting - key takeaways

- Always be careful when downloading packages, be precise about spelling, and never guess a package name
- Typosquatting can inject malicious packages through indirect dependencies, which can be hard to spot
- Keep an eye on your dependency tree, it is important to know what you are using so you can spot problems when they occur