Распутываем зависимости

и прочие неурядицы

Спикеры

Денис Тарасов

Тимлид

Дмитрий Афанасьев

Старший инженер

01

Артефакты

Что такое артефакты?

Вспомогательные элементы продукта, так или иначе входящие в его состав.

Что такое артефакты?

Вспомогательные элементы продукта, так или иначе входящие в его состав.

Например:

- Исполняемые файлы

Что такое артефакты?

Вспомогательные элементы продукта, так или иначе входящие в его состав.

Например:

- Исполняемые файлы
- Библиотеки

02

Зависимости

Что такое зависимость?

Зависимость - это объект, который может быть использован как сервис.

Зависимости могут быть как внутри приложения, так и внешние библиотеки

Что такое версионирование?

Нумерация версий программного обеспечения, процесс присвоения уникальных имен версий или номеров уникальным состояниям компьютерного программного обеспечения.

Что такое версионирование?

Нумерация версий программного обеспечения, процесс присвоения уникальных имен версий или номеров уникальным состояниям компьютерного программного обеспечения.

Например:

- MyCompany.Core-1.0.0.nupkg
- MyCompany.Core-1.0.1.nupkg

Что такое SEMVER?

Спецификация о том, как присваивать версии релизам программного обеспечения

https://semver.org/

Что такое SEMVER?

Спецификация о том, как присваивать версии релизам программного обеспечения

- {major}.

https://semver.org/

Что такое SEMVER?

Спецификация о том, как присваивать версии релизам программного обеспечения

- {major}.
- {minor}.

https://semver.org/

Что такое SEMVER?

Спецификация о том, как присваивать версии релизам программного обеспечения

- {major}.
- {minor}.
- {patch}

https://semver.org/

03

Цепочка поставок

Что такое цепочка поставок?

Это система ресурсов, участвующая в жизненном цикле продукта, от момента его проектирования до поставки конечному пользователю

Artifact

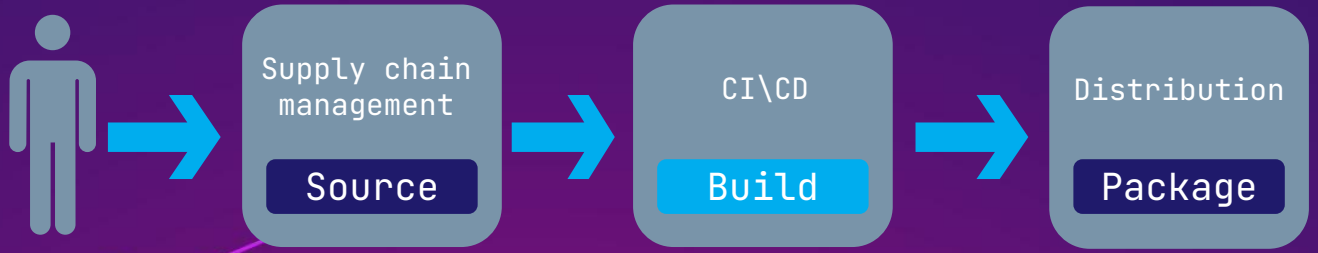
Process

Platform

https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html

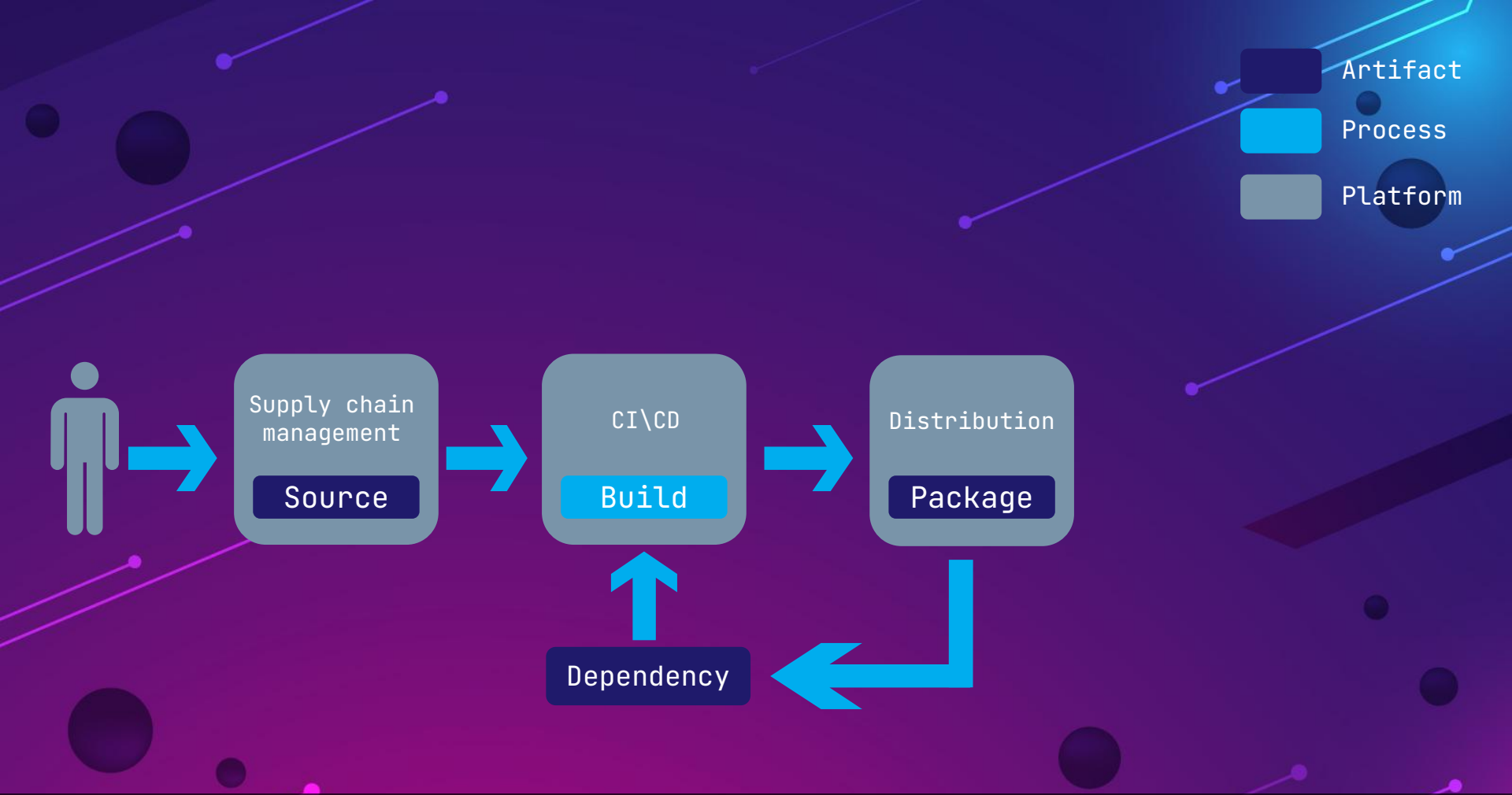
Artifact
Process
Platform

https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html

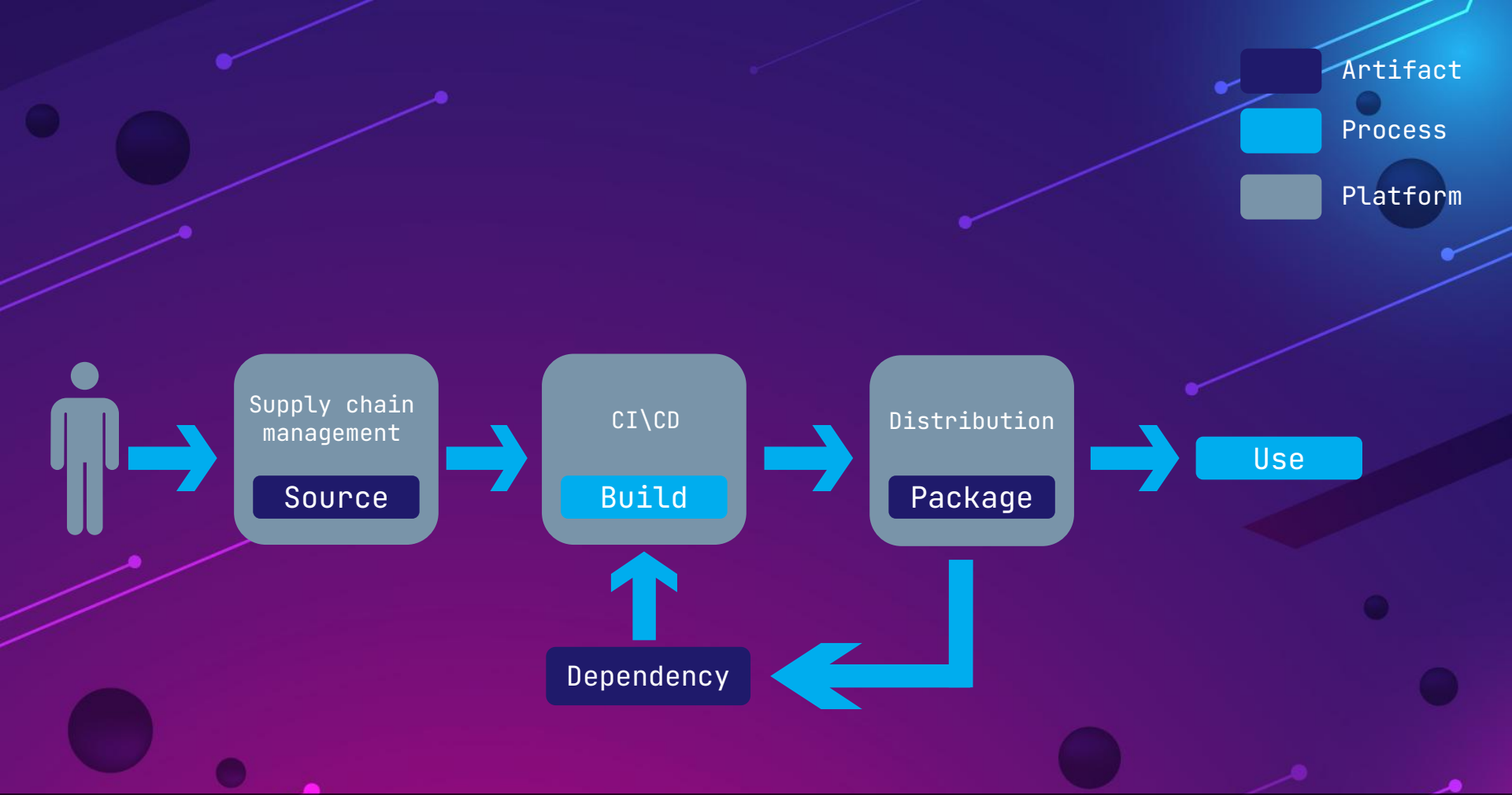


Artifact
Process
Platform

https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html

04

Атака на цепочку

Что такое атака на цепочку поставок?

Атакой на цепочку поставок называют взлом устройства или компьютерной программы, совершенный до момента передачи их потребителю

Что такое атака на цепочку поставок?

Атакой на цепочку поставок называют взлом устройства или компьютерной программы, совершенный до момента передачи их потребителю

- Внедрение вредоносного кода в OSS-библиотеку

Что такое атака на цепочку поставок?

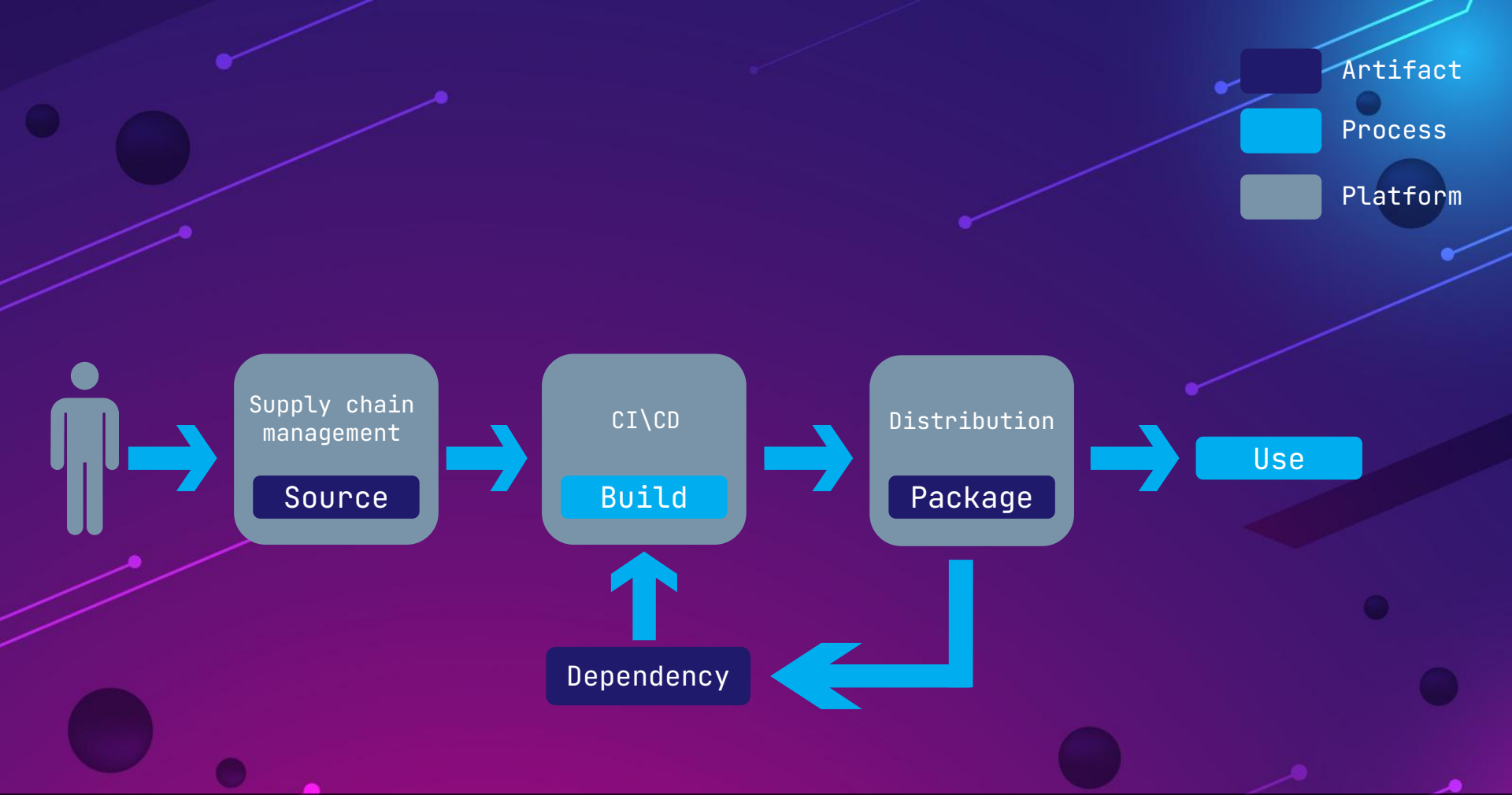
Атакой на цепочку поставок называют взлом устройства или компьютерной программы, совершенный до момента передачи их потребителю

- Внедрение вредоносного кода в OSS-библиотеку
- Предварительно установленное вредоносное ПО на устройстве (телефоне, камере, ПК)

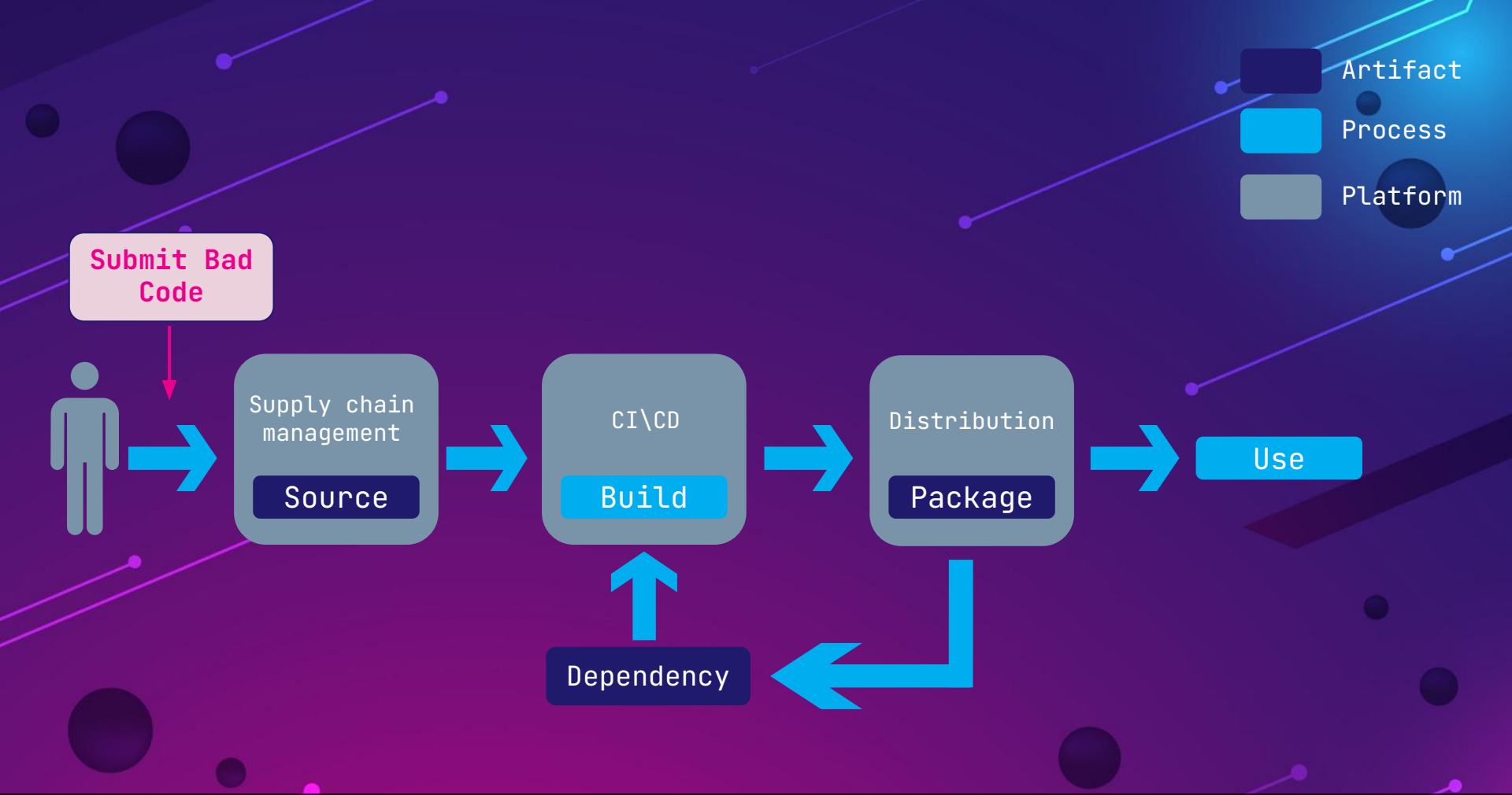
Что такое атака на цепочку поставок?

Атакой на цепочку поставок называют взлом устройства или компьютерной программы, совершенный до момента передачи их потребителю

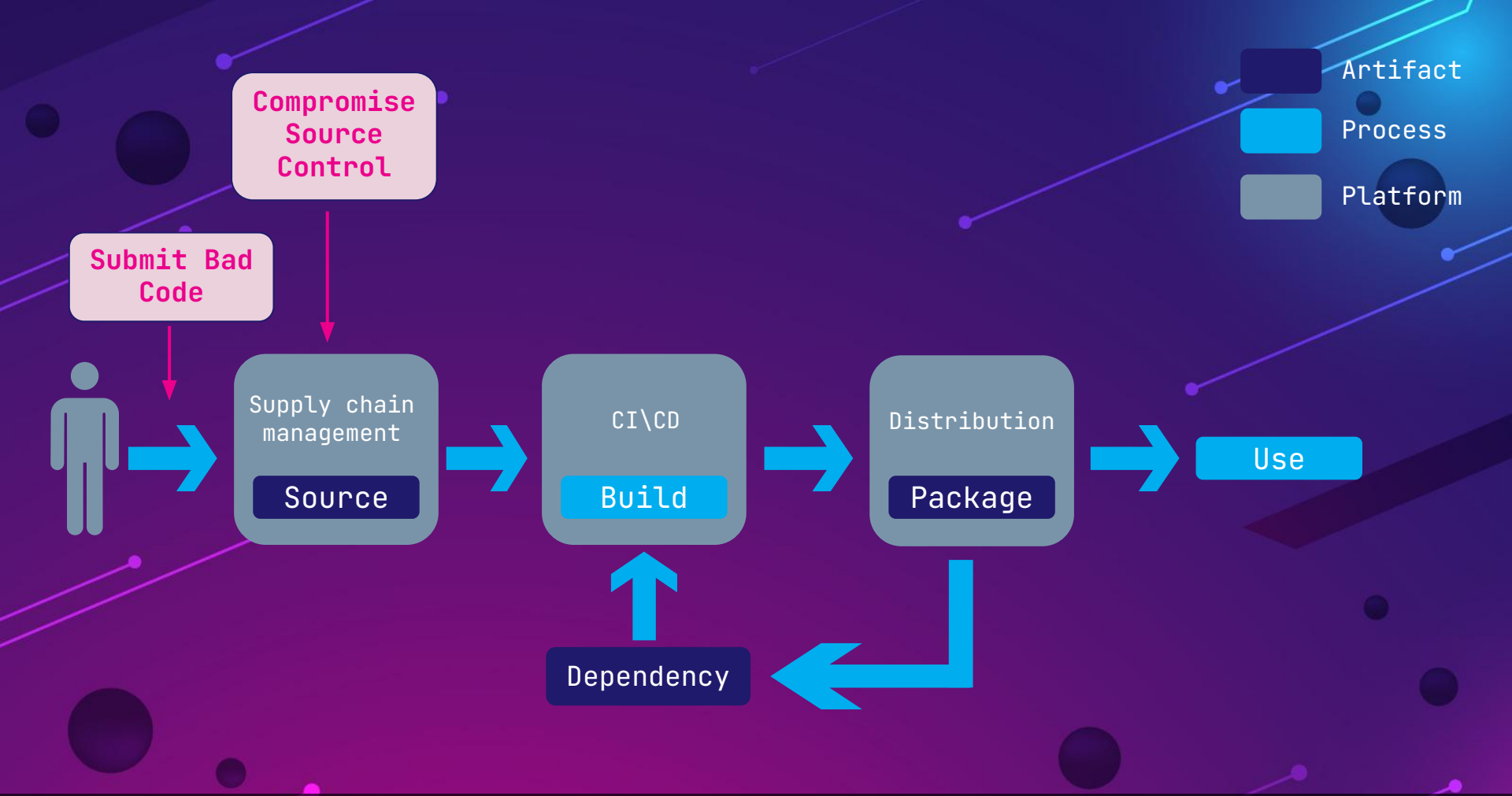
- Внедрение вредоносного кода в OSS-библиотеку
- Предварительно установленное вредоносное ПО на устройстве (телефоне, камере, ПК)
- Скомпрометированные инструменты сборки и деплоя



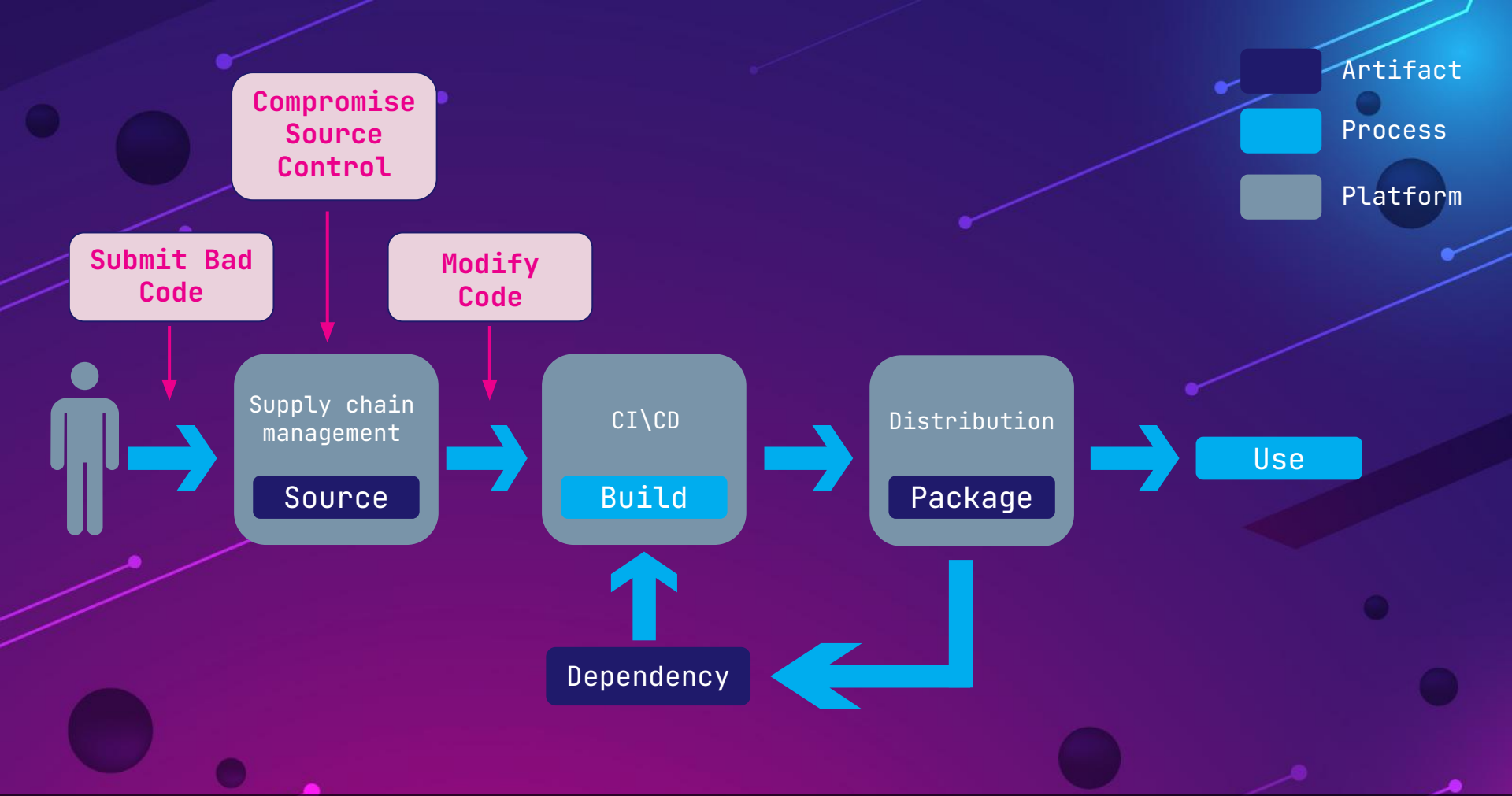
https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



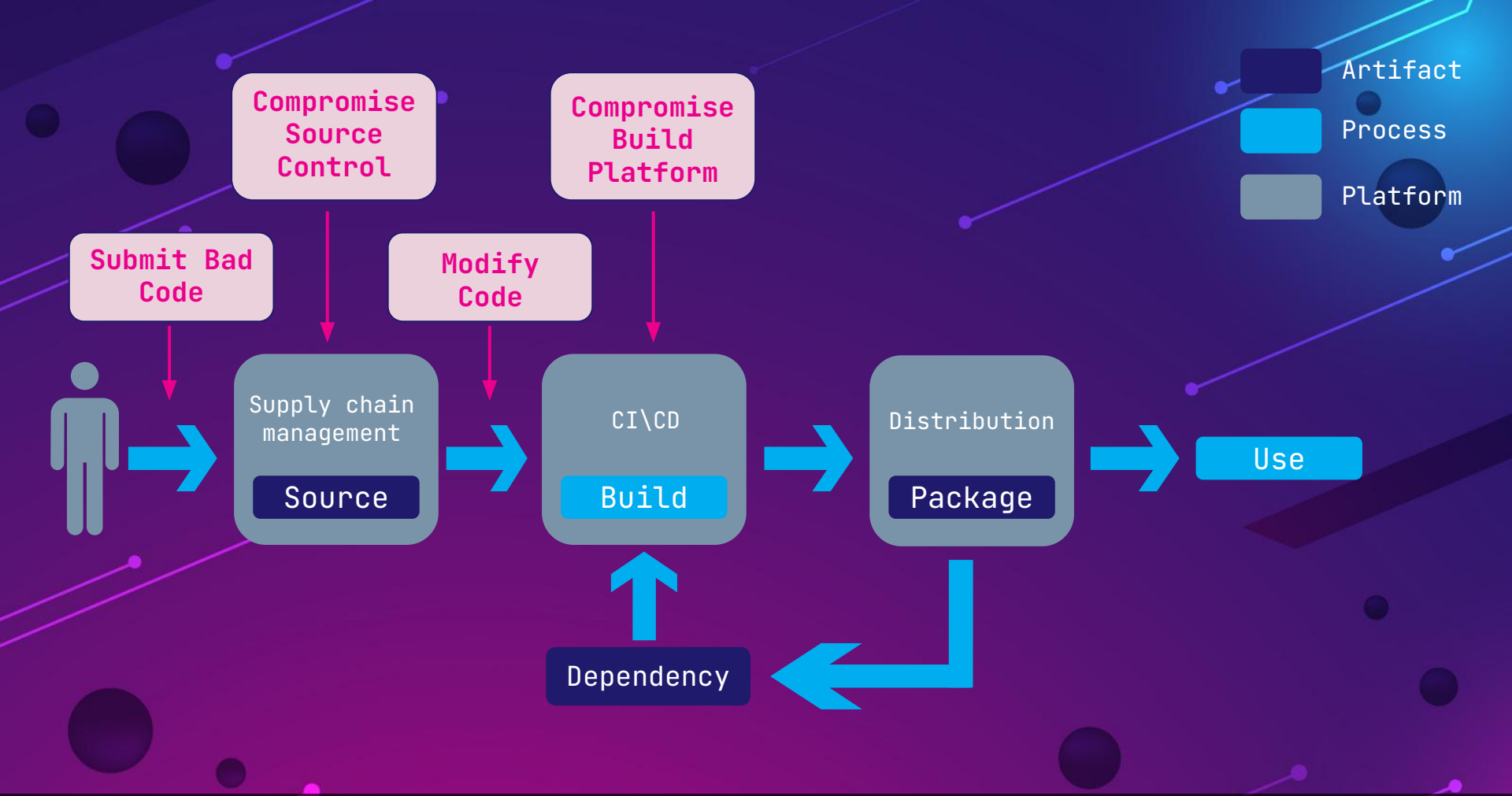
https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



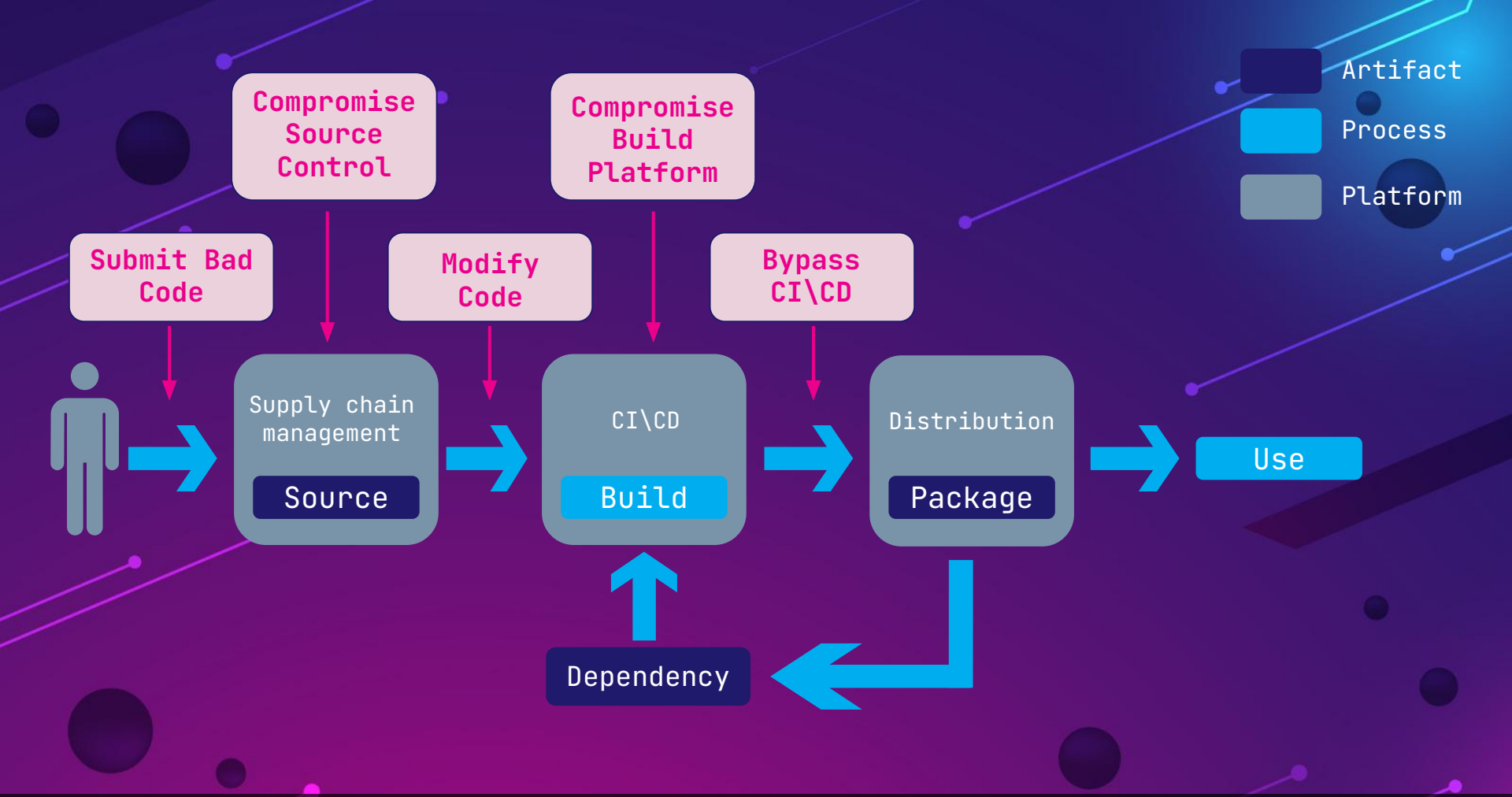
https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



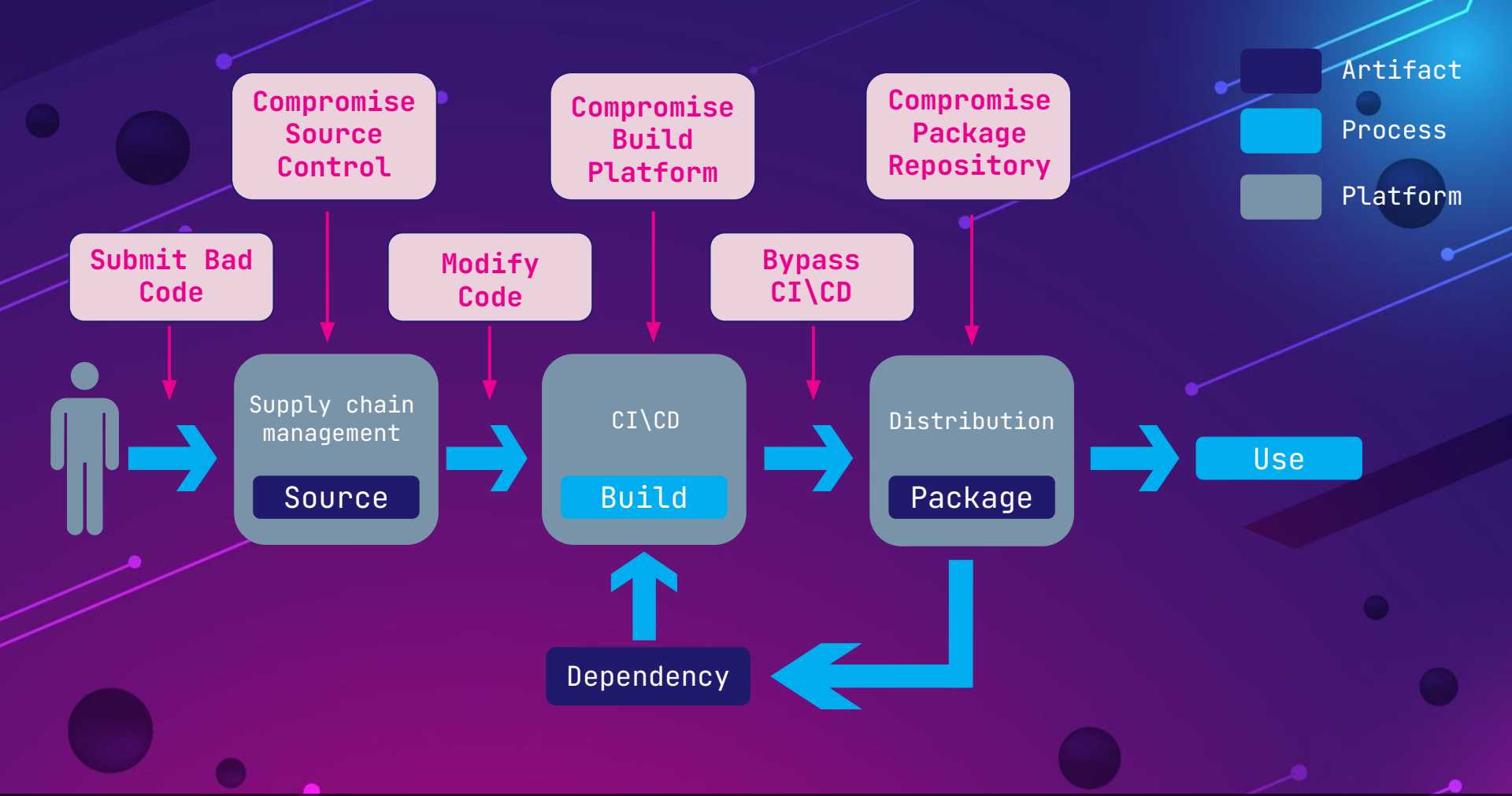
https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



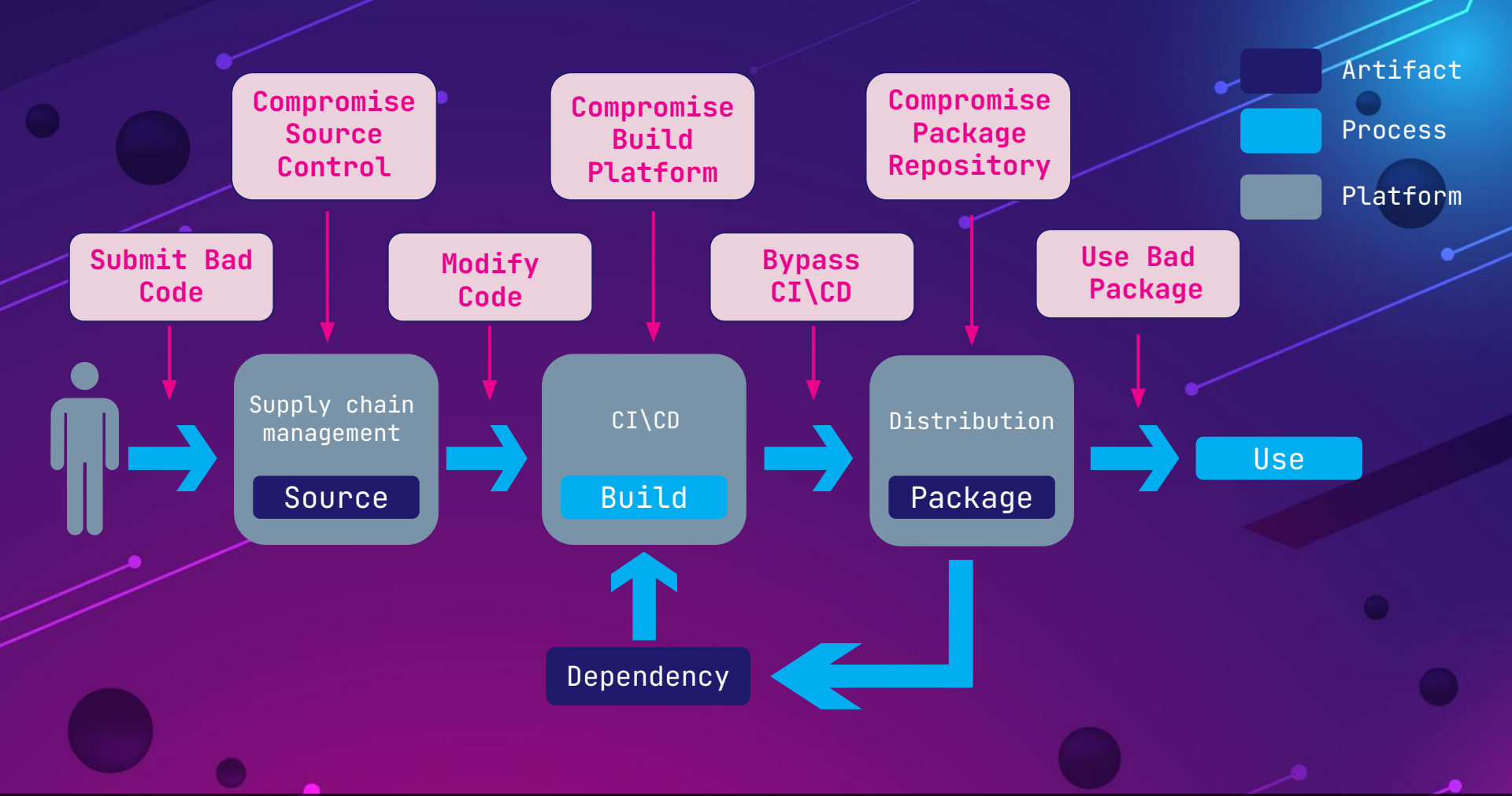
https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



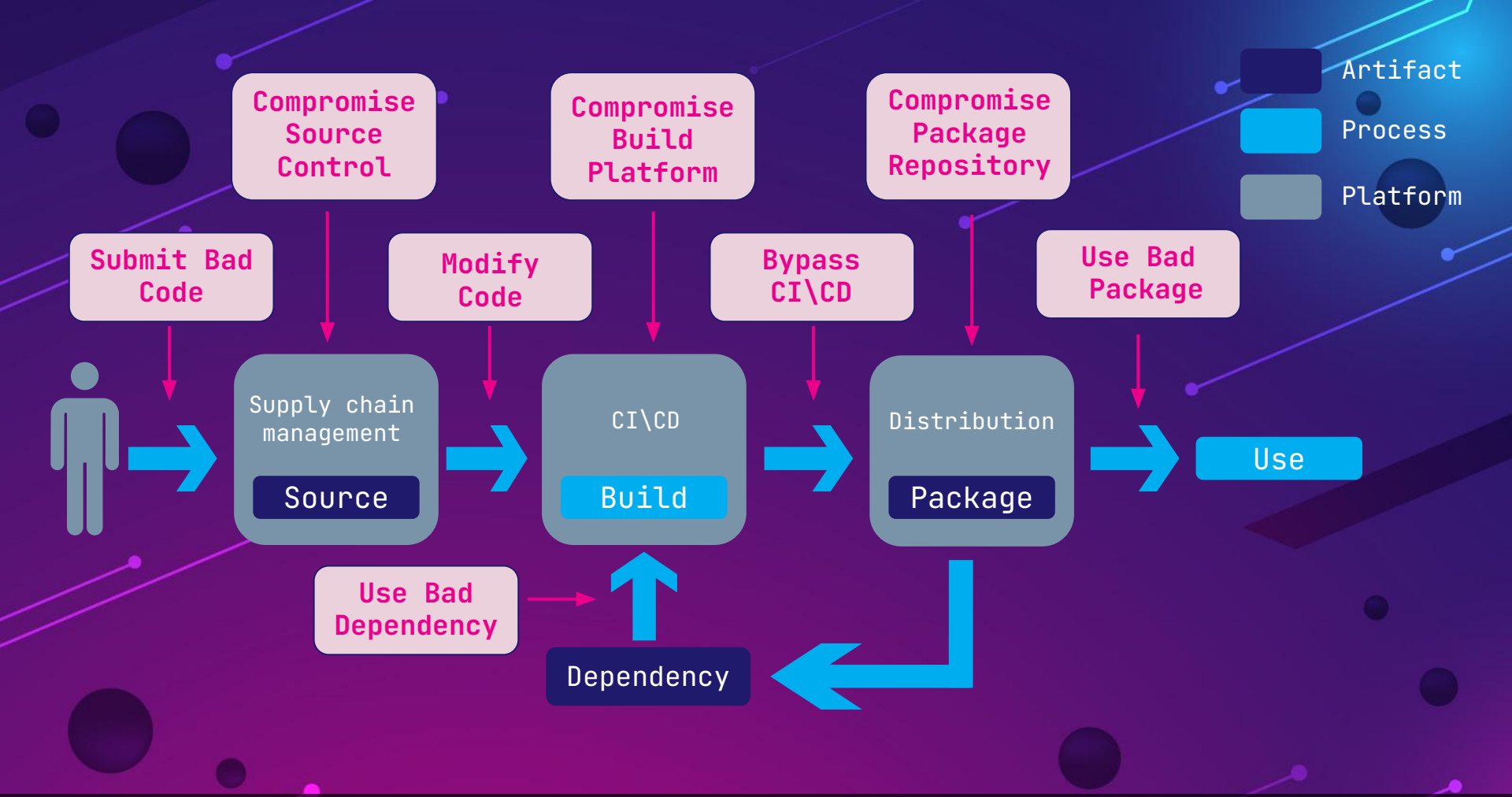
https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html



https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html

05

Compromise build platform

solarwinds

The Power to Manage IT

Ключевые события в атаке

01

Сентябрь 2019

Злоумышленники получили доступ к сети

Ключевые события в атаке

01

Сентябрь 2019

Злоумышленники получили доступ к сети

02

Октябрь 2019

Первоначальная инъекция вредоносного кода

Ключевые события в атаке

01

Сентябрь 2019

Злоумышленники получили доступ к сети

02

Октябрь 2019

Первоначальная инъекция вредоносного кода

03

Февраль 2020

Внедрен вредоносный код, известный как Sunbrust

Ключевые события в атаке

01

Сентябрь 2019

Злоумышленники получили доступ к сети

02

Октябрь 2019

Первоначальная инъекция вредоносного кода

03

Февраль 2020

Внедрен вредоносный код, известный как Sunbrust

04

Март 2020

Рассылка обновлений с вредоносным кодом

Пострадавшие

Dep of treasure

Пострадавшие

Dep of treasure

Dep of state

Пострадавшие

Dep of treasure

Dep of state

Dep of energy

Пострадавшие

Dep of treasure

Dep of state

Dep of energy

Nuclear regulatory

Пострадавшие

Dep of treasure

Dep of state

Dep of energy

Nuclear regulatory

Microsoft

Пострадавшие

Dep of treasure

Dep of state

Dep of energy

Nuclear regulatory

Microsoft

Fire eye

Последствия

- Утекла переписка сотрудников

Последствия

- Утекла переписка сотрудников
- Утекли клиентские учетные данные

Последствия

- Утекла переписка сотрудников
- Утекли клиентские учетные данные
- Скомпрометирована не только компания, но и клиенты

Последствия

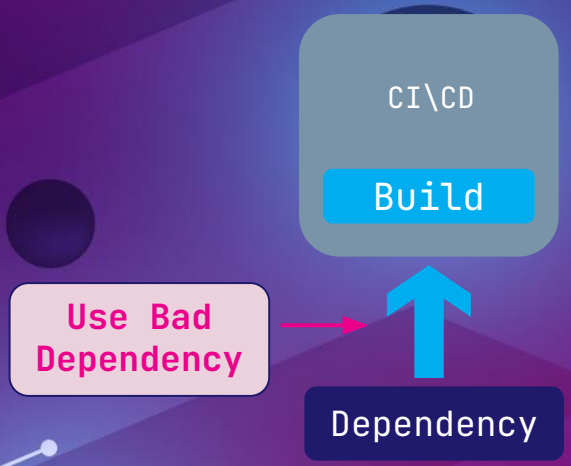
- Утекла переписка сотрудников
- Утекли клиентские учетные данные
- Скомпрометирована не только компания, но и клиенты
- Пострадало около 18 тысяч клиентов

Последствия

- Утекла переписка сотрудников
- Утекли клиентские учетные данные
- Скомпрометирована не только компания, но и клиенты
- Пострадало около 18 тысяч клиентов
- Финансовые потери ~ 90 млн долларов

Последствия

- Утекла переписка сотрудников
- Утекли клиентские учетные данные
- Скомпрометирована не только компания, но и клиенты
- Пострадало около 18 тысяч клиентов
- Финансовые потери ~ 90 млн долларов
- Репутационные потери



06

Use bad dependency

Log4j2 - CVE-2021-44228

Apache Log4j2 2.0-beta9 through 2.15.0 (excluding security releases 2.12.2, 2.12.3, and 2.3.1) JNDI features used in configuration, log messages, and parameters do not protect against attacker controlled LDAP and other JNDI related endpoints.

An attacker who can control log messages or log message parameters can execute arbitrary code loaded from LDAP servers when message lookup substitution is enabled

Severity

Critical 10.0 / 10

CVSS base metrics

Attack vector	Network
Attack complexity	Low
Privileges required	None
User interaction	None
Scope	Changed
Confidentiality	High
Integrity	High
Availability	High

https://logging.apache.org/log4j/2.x/security.html

Node-ipc - CVE-2022-23812

The package node-ipc versions 10.1.1 and 10.1.2 are vulnerable to embedded malicious code that was introduced by the maintainer.

The malicious code was intended to overwrite arbitrary files dependent upon the geo-location of the user IP address.

The maintainer removed the malicious code in version 10.1.3.

Severity

Critical 9.8 / 10

CVSS base metrics

Attack vector	Network
Attack complexity	Low
Privileges required	None
User interaction	None
Scope	Unchanged
Confidentiality	High
Integrity	High
Availability	High

https://github.com/advisories/GHSA-97m3-w2cp-4xx6

07

Dependency confusion

Alex Birsan

https://medium.com/@alex.birsan

https://twitter.com/alxbrsn

https://hackerone.com/alexbirsan

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Dependency confusion

Netflix

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Dependency confusion

Netflix

Paypal

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Dependency confusion

Netflix

Paypal

Apple

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Dependency confusion

Netflix

Paypal

Apple

Microsoft

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Dependency confusion

```
"dependencies": {
  "express": "^4.3.0",
  "dustjs-helpers": "~1.6.3",
  "continuation-local-storage": "^3.1.0",
  "pplogger": "^0.2",
  "auth-paypal": "^2.0.0",
  "wurfl-paypal": "^1.0.0",
  "analytics-paypal": "~1.0.0"
}
```

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Dependency confusion

- Опубликовал npm, pip-пакеты и ruby-gems с такими же именами, но уже с “вредоносным” кодом

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Dependency confusion

- Опубликовал npm, pip-пакеты и ruby-gems с такими же именами, но уже с “вредоносным” кодом
- При сборках компании вытягивали “неправильные” пакеты

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Dependency confusion

- Опубликовал npm, pip-пакеты и ruby-gems с такими же именами, но уже с “вредоносным” кодом
- При сборках компании вытягивали “неправильные” пакеты
- В общей сложности ему выплатили около 140 тыс. долларов по Bug Bounty

https://medium.com/@alex.birsan/dependency-confusion-4a5d60fec610

Как же так вышло?

Npm scripts

```
{
  "scripts" : {
    "install" : "scripts/install.js",
    "postinstall" : "scripts/install.js",
    "uninstall" : "scripts/uninstall.js"
  }
}
```

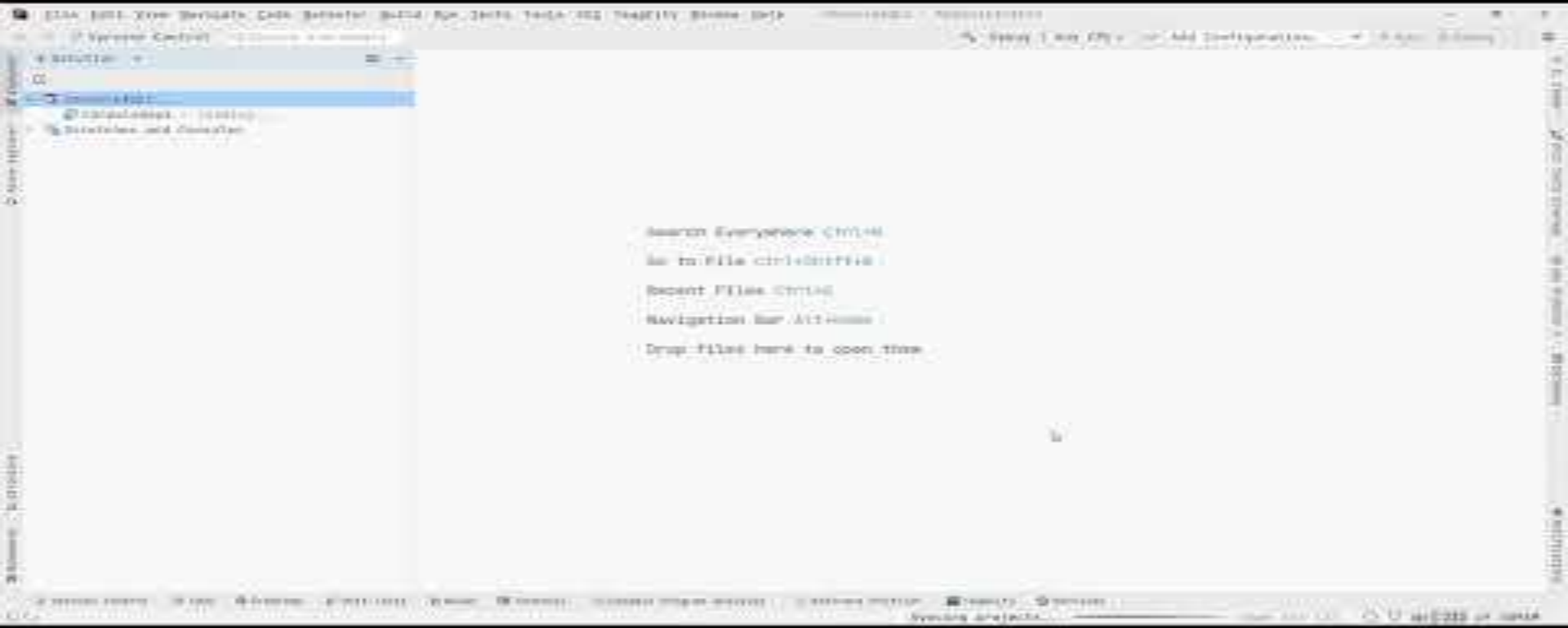
Nuget

packages.config

- install.ps1
- uninstall.ps1

https://docs.microsoft.com/en-us/nuget/reference/packages-config

А ты запускаешь IDE как администратор?



https://www.youtube.com/watch?v=4jUAsZ516fE

AAAAAAAAAAAAAAA!!!

08

Вектор атаки

Используете Docker desktop?

ILSpy

File View Window Help

C# | C# 10.0 / VS 2022

Assemblies

- mscorlib (4.0.0.0, .NETFramework, v4.0)
- System (4.0.0.0, .NETFramework, v4.0)
- System.Core (4.0.0.0, .NETFramework, v4.0)
- System.Xml (4.0.0.0, .NETFramework, v4.0)
- System.Xaml (4.0.0.0, .NETFramework, v4.0)
- WindowsBase (4.0.0.0, .NETFramework, v4.0)
- PresentationCore (4.0.0.0, .NETFramework, v4.0)
- PresentationFramework (4.0.0.0, .NETFramework, v4.0)
- Docker Desktop (1.0.0.0, .NETFramework, v4.6.2)

Docker Desktop

```
// C:\Program Files\Docker\Docker\Docker Desktop.exe
// Docker Desktop, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null
// Global type: <Module>
// Entry point: Docker.Program.Main
// Architecture: x64
// Runtime: v4.0.30319
// This assembly was compiled using the /deterministic option.
// Hash algorithm: SHA1
// Debug info: Loaded from PDB file: C:\Program Files\Docker\Docker\Docker Desktop.pdb

using ...

[assembly: CompilationRelaxations(8)]
[assembly: RuntimeCompatibility(WrapNonExceptionThrows = true)]
[assembly: Debuggable(DebuggableAttribute.DebuggingModes.IgnoreSymbolStoreSequencePoints)]
[assembly: InternalsVisibleTo("Docker.Desktop.Tests")]
[assembly: AssemblyDescription("Docker Desktop")]
[assembly: ComVisible(false)]
[assembly: Guid("febed07f-9df0-4741-9526-58c6429a384f")]
[assembly: TargetFramework(".NETFramework,Version=v4.6.2", FrameworkDisplayName = ".NET Framework 4.6.2")]
[assembly: AssemblyCompany("Docker Desktop")]
[assembly: AssemblyConfiguration("Release")]
[assembly: AssemblyFileVersion("1.0.0.0")]
[assembly: AssemblyInformationalVersion("1.0.0")]
[assembly: AssemblyProduct("Docker Desktop")]
[assembly: AssemblyTitle("Docker Desktop")]
[assembly: AssemblyVersion("1.0.0.0")]
```

Name	Date modified	Type	Size
frontend	24.08.2022 18:55	File folder	
resources	24.08.2022 18:56	File folder	
app.json	24.08.2022 18:54	JSON File	5 KB
BITSReference5_0.dll	24.08.2022 18:55	Application exten...	21 KB
Bugsnag.dll	24.08.2022 18:55	Application exten...	70 KB
com.docker.service	24.08.2022 18:54	SERVICE File	20 KB
com.docker.service.config	24.08.2022 18:54	Configuration Sou...	20 KB
com.docker.service.pdb	24.08.2022 18:54	VisualStudio.pdb....	28 KB
courgette64.exe	24.08.2022 18:55	Application	944 KB
Docker Desktop Installer.exe	24.08.2022 18:55	Application	7 224 KB
Docker Desktop Installer.exe.config	24.08.2022 18:54	Configuration Sou...	20 KB
Docker Desktop Installer.pdb	24.08.2022 18:54	VisualStudio.pdb....	370 KB
Docker Desktop.exe	24.08.2022 18:55	Application	275 KB
Docker Desktop.exe.config	24.08.2022 18:54	Configuration Sou...	20 KB
Docker Desktop.pdb	24.08.2022 18:54	VisualStudio.pdb....	118 KB
Docker.ApiServices.dll	24.08.2022 18:55	Application exten...	152 KB
Docker.ApiServices.pdb	24.08.2022 18:54	VisualStudio.pdb....	592 KB
Docker.Backend.dll	24.08.2022 18:55	Application exten...	97 KB
Docker.Backend.pdb	24.08.2022 18:54	VisualStudio.pdb....	236 KB
Docker.Core.dll	24.08.2022 18:55	Application exten...	178 KB
Docker.Core.pdb	24.08.2022 18:54	VisualStudio.pdb....	610 KB
Docker.Engines.dll	24.08.2022 18:55	Application exten...	109 KB
Docker.Engines.pdb	24.08.2022 18:54	VisualStudio.pdb....	264 KB
Docker.HttpApi.dll	24.08.2022 18:55	Application exten...	34 KB

docker.engines

0 packages returned for docker.engines

Filter

NuGet package search works the same on nuget.org, from the NuGet CLI, and within the NuGet Package Manager extension in Visual Studio.
Check out our Search Syntax.

Contact

Got questions about NuGet or the NuGet Gallery?

Status

Find out the service status of NuGet.org and its related services.

FAQ

Read the Frequently Asked Questions about NuGet and see if your question made the list.

Пакет-то приватный!

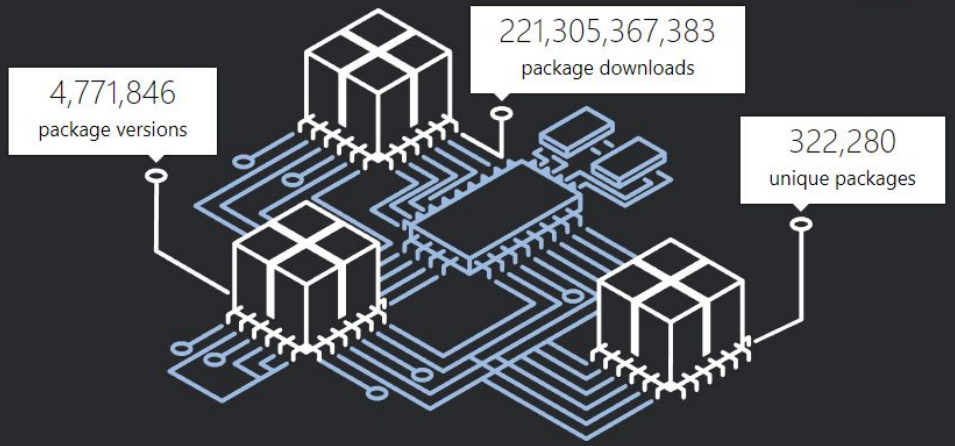
09

Artifact feeds

nuget Packages Upload Statistics Documentation Downloads Blog Sign in

Create .NET apps faster with NuGet

Search for packages...



What is NuGet?

NuGet is the package manager for .NET. The NuGet client tools provide the ability to produce and consume packages. The NuGet Gallery is the central package repository used by all package authors and consumers.

Sonatype Nexus Repository Manager
OSS 3.38.1-01

Search components

Sign in

Browse

- Welcome
- Search
- Browse

Welcome Learn about Sonatype Nexus Repository Manager

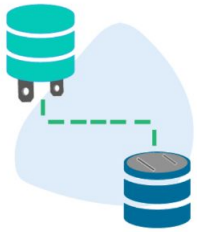
What's new in Nexus Repository 3.40 Pro?

New repository connector type!

In this version, we introduce subdomain routing for Docker repositories.

With subdomain routing, you no longer need to use port connectors or remember a lengthy list of port numbers. Create more easily memorable subdomains with logically assigned names instead.

You'll also experience the added benefit of avoiding the performance limitations that come with port connectors. Learn about these changes in our help documentation and check out the "Have you heard" video.



Open Source Attacks on the Rise: Top 8 Malicious Packages Found in npm

Read More...

Get Started

Upgrading
Upgrade to the latest version

Configuration
Set things up properly

Documentation
Visit our help site

Help us understand your needs!

Let us get to know you!

English

* 1. Which of the following is the closest to your current role?

- Developer/Engineer
- Engineering Manager
- Administrator
- Application Security
- Other (please specify)

10

Приватные пакеты и репозитории

Типовой nuget.config

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <packageSources>
    <add key="MyCompany feed" value="https://nuget.mycompany.com/v3/index.json">
   </packageSources>
</configuration>
```

Откуда будет качаться пакет mycompany.core?

Откуда будет качаться пакет mycompany.core?

nuget.org

Откуда будет качаться пакет mycompany.core?

nuget.org

nuget.mycompany.com

Откуда будет качаться пакет mycompany.core?

?

nuget.org

nuget.mycompany.com

11

Порядок применения nuget.config

Порядок применения nuget.config

https://docs.microsoft.com/ru-ru/nuget/consume-packages/configuring-nuget-behavior

Порядок применения nuget.config

1. NuGetDefaults.Config file

OS Platform	`NuGetDefaults.Config` Location
Windows	Visual Studio 2017 or NuGet 4.x+: `%ProgramFiles(x86)%\NuGet\Config` Visual Studio 2015 and earlier or NuGet 3.x and earlier: `%PROGRAMDATA%\NuGet`
Mac/Linux	`$XDG_DATA_HOME` (typically `~/.local/share` or `/usr/local/share`, depending on OS distribution)

https://docs.microsoft.com/ru-ru/nuget/consume-packages/configuring-nuget-behavior

Порядок применения nuget.config

1. NuGetDefaults.Config file
2. computer-level file

Computer	**Windows:** `%ProgramFiles(x86)%\NuGet\Config` **Mac/Linux:** `$XDG_DATA_HOME`. If `$XDG_DATA_HOME` is null or empty, `~/.local/share` or `/usr/local/share` will be used (varies by OS distribution)	Settings apply to all operations on the computer, but are overridden by any user- or project-level settings.

https://docs.microsoft.com/ru-ru/nuget/consume-packages/configuring-nuget-behavior

Порядок применения nuget.config

1. NuGetDefaults.Config file
2. computer-level file

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <packageSources>
    <add key="nuget.org" value="https://api.nuget.org/v3/index.json">
   </packageSources>
</configuration>
```

https://docs.microsoft.com/ru-ru/nuget/consume-packages/configuring-nuget-behavior

Порядок применения nuget.config

1. NuGetDefaults.Config file
2. computer-level file
3. user-level file

User	**Windows:** `%appdata%\NuGet\NuGet.Config` **Mac/Linux:** `~/.config/NuGet/NuGet.Config` or `~/.nuget/NuGet/NuGet.Config` (varies by OS distribution) Additional configs are supported on all platforms. These configs cannot be edited by the tooling. **Windows:** `%appdata%\NuGet\config\*.Config` **Mac/Linux:** `~/.config/NuGet/config/*.config` or `~/.nuget/config/*.config`	Settings apply to all operations, but are overridden by any project-level settings.

https://docs.microsoft.com/ru-ru/nuget/consume-packages/configuring-nuget-behavior

Порядок применения nuget.config

1. NuGetDefaults.Config file
2. computer-level file
3. user-level file
4. the file specified with -configfile

https://docs.microsoft.com/ru-ru/nuget/consume-packages/configuring-nuget-behavior

Порядок применения nuget.config

1. NuGetDefaults.Config file
2. computer-level file
3. user-level file
4. the file specified with -configfile
5. Files found in every folder in the path from the drive root to the current folder

https://docs.microsoft.com/ru-ru/nuget/consume-packages/configuring-nuget-behavior

Список
<packageSources>
собирается из всех
конфигов!

Порядок источников в packageSources важен!

</clear>

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <packageSources>
    </clear>
    <add key="MyComnpanyFeed" value="https://nuget.mycompany.com/v3/index.json">
    <add key="nuget.org" value="https://api.nuget.org/v3/index.json">
   </packageSources>
</configuration>
```

Добавляйте nuget.config в репозитории!

Откуда будет качаться пакет mycompany.core?

Откуда будет качаться пакет mycompany.core?

?

nuget.org

nuget.mycompany.com

Откуда будет качаться пакет mycompany.core?

nuget.org

Зависит

nuget.mycompany.com

А где опубликован mycompany.core?

nuget.mycompany.com

```
MyCompany.Extensions 1.0.0
MyCompany.Core 1.0.5
```

nuget.org

```
Serilog 2.11.0
Newtonsoft.Json 13.0.1
MyCompany.Core 1.0.6
```

12

Nexus repository manager

Типы репозиториев

Hosted

Типы репозиториев

Hosted

Proxy
nuget.org

Типы репозиториев

Hosted

Proxy
nuget.org

Group

Hosted

Proxy
nuget.org

Итоговая схема работы

Group

Hosted

Proxy
nuget.org

Итоговый nuget.config

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <packageSources>
    </clear>
    <add key="MyComnpanyFeed" value="https://nuget.mycompany.com/v3/index.json">
   </packageSources>
</configuration>
```

Откуда скачается mycompany.core

Group

Hosted

Proxy
nuget.org

Откуда скачается mycompany.core

Group

Hosted

Proxy
nuget.org

13

Nexus routing rules

Nexus routing rules

Sonatype Nexus Repository Manager
OSS 3.38.1-01

Search components

Administration

- Repository
  - Repositories
  - Blob Stores
  - Cleanup Policies
  - Content Selectors
  - Log4j Visualizer
  - Proprietary Repositories
  - Routing Rules
- Security
  - Privileges
  - Roles
  - Users
  - Anonymous Access
  - LDAP
  - Realms
  - SSL Certificates
- IQ Server
- Support
  - Logging
    - Log Viewer

Name

Block_MyCompany_From_Proxy

Description *Optional*

Mode

Allow or block requests when their path matches any of the following matchers

Block

Matchers

Enter regular expressions that will be used to identify request paths to allow or block (depending on above mode)

(.*)\/((MyCompany\.)(.*))

+ Add Another Matcher

Path

Enter a request path to check if it would be blocked or allowed. Requests always start with a leading slash.

/ repository/nuget/v3/content/MyComp

Test

This request would be blocked

https://help.sonatype.com/repomanager3/nexus-repository-administration/repository-management/routing-rules

Nexus routing rules

```
(.*)\/(MyCompany\.(.*))
```

https://help.sonatype.com/repomanager3/nexus-repository-administration/repository-management/routing-rules

Мы в безопасности?

Откуда будет качаться пакет mycompany.core?

Group

Hosted

Proxy
nuget.org

Откуда будет качаться пакет mycompany.core?

Group

Hosted

Proxy
nuget.org

Откуда будет качаться пакет mycompany.core?

Group

Hosted

Proxy
nuget.org

Откуда будет качаться пакет mycompany.core?

Group

Hosted

Proxy
nuget.org

А может скачаться еще откуда-то?

Откуда будет качаться пакет mycompany.core?

Cache

Group

Hosted

Proxy
nuget.org

Nuget cache

1. global packages
2. http-cache
3. temp
4. plugins-cache (nuget 4.8+)

https://docs.microsoft.com/en-us/nuget/consume-packages/managing-the-global-packages-and-cache-folders

Nuget cache

cli

Copy

```cli
# Clear the 3.x+ cache (use either command)
dotnet nuget locals http-cache --clear
nuget locals http-cache -clear

# Clear the 2.x cache (NuGet CLI 3.5 and earlier only)
nuget locals packages-cache -clear

# Clear the global packages folder (use either command)
dotnet nuget locals global-packages --clear
nuget locals global-packages -clear

# Clear the temporary cache (use either command)
dotnet nuget locals temp --clear
nuget locals temp -clear

# Clear the plugins cache (use either command)
dotnet nuget locals plugins-cache --clear
nuget locals plugins-cache -clear

# Clear all caches (use either command)
dotnet nuget locals all --clear
nuget locals all -clear
```

https://docs.microsoft.com/en-us/nuget/consume-packages/managing-the-global-packages-and-cache-folders

Nuget cache

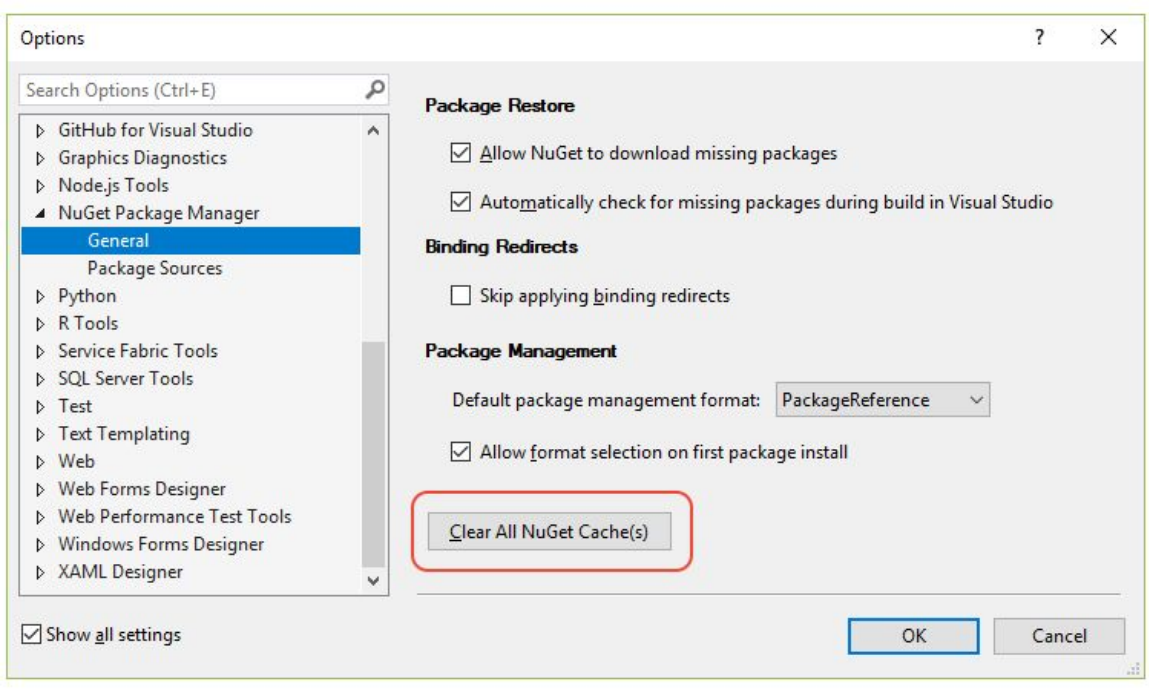
cli

Copy

```cli
# Clear the 3.x+ cache (use either command)
dotnet nuget locals http-cache --clear
nuget locals http-cache -clear

# Clear the 2.x cache (NuGet CLI 3.5 and earlier only)
nuget locals packages-cache -clear

# Clear the global packages folder (use either command)
dotnet nuget locals global-packages --clear
nuget locals global-packages -clear

# Clear the temporary cache (use either command)
dotnet nuget locals temp --clear
nuget locals temp -clear

# Clear the plugins cache (use either command)
dotnet nuget locals plugins-cache --clear
nuget locals plugins-cache -clear

# Clear all caches (use either command)
dotnet nuget locals all --clear
nuget locals all -clear
```

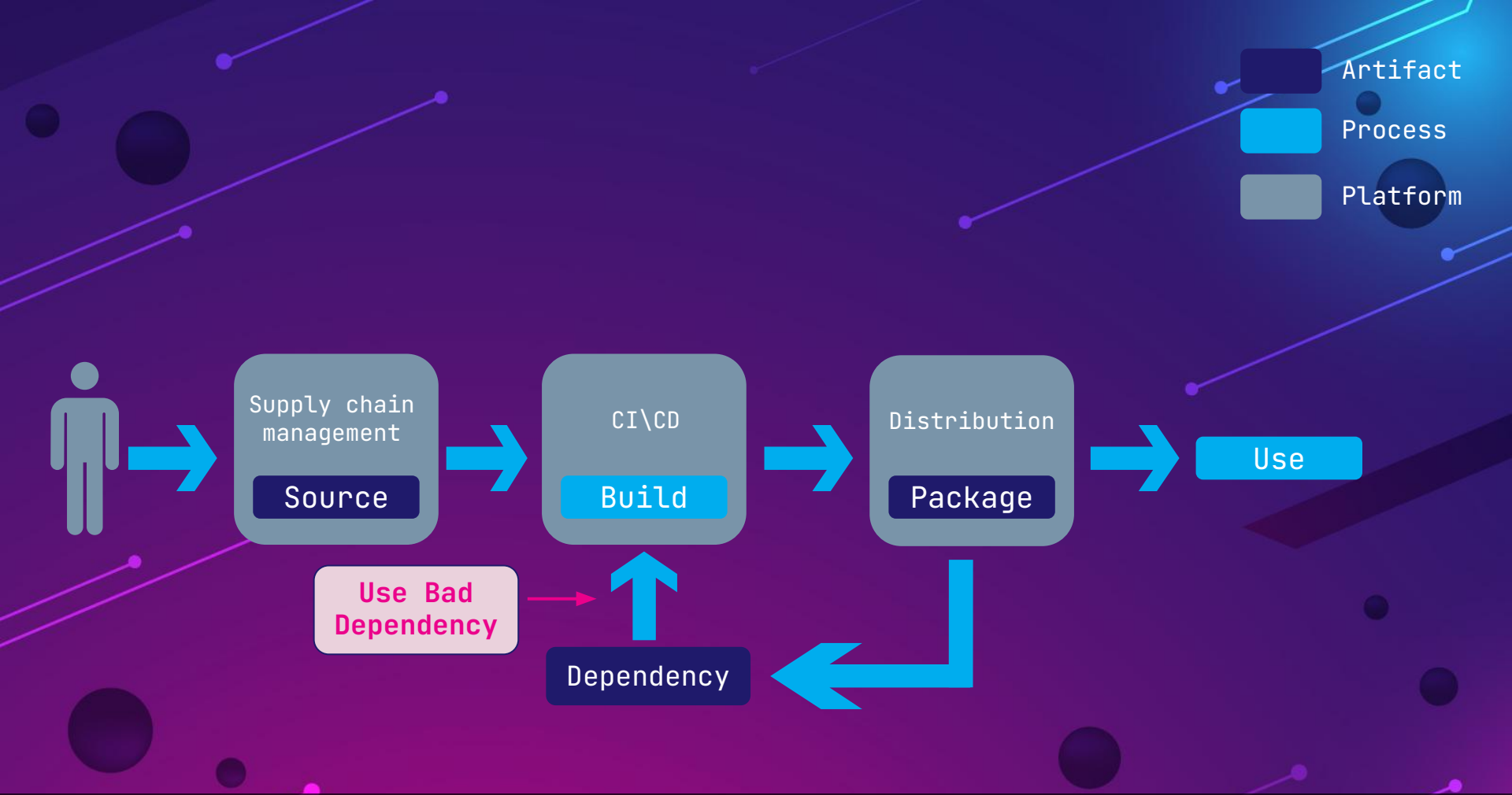
https://docs.microsoft.com/en-us/nuget/consume-packages/managing-the-global-packages-and-cache-folders

Nuget cache



https://docs.microsoft.com/en-us/nuget/consume-packages/managing-the-global-packages-and-cache-folders

Мы в безопасности?



https://opensource.googleblog.com/2021/10/protect-your-open-source-project-from-supply-chain-attacks.html

Что может пойти не так?

Что может пойти не так?

1. Собирается проект 1

Что может пойти не так?

1. Собирается проект 1 - OK

Что может пойти не так?

1. Собирается проект 1 - OK
2. Кто-то почистил кэш

Что может пойти не так?

1. Собирается проект 1 - OK
2. Кто-то почистил кэш
3. Собирается проект 2

Что может пойти не так?

1. Собирается проект 1 - OK
2. Кто-то почистил кэш
3. Собирается проект 2 - POISONED

Что может пойти не так?

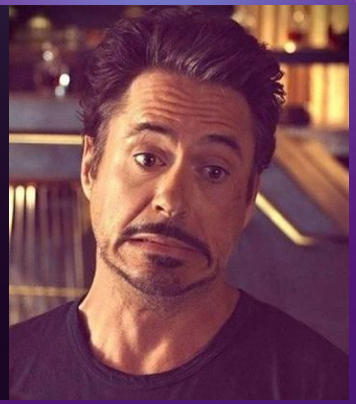
1. Собирается проект 1 - OK
2. Кто-то почистил кэш
3. Собирается проект 2 - POISONED
4. Собирается проект 1

Что может пойти не так?

1. Собирается проект 1 - OK
2. Кто-то почистил кэш
3. Собирается проект 2 - POISONED
4. Собирается проект 1 - POISONED

Что может пойти не так?

1. Собирается проект 1 - OK
2. Кто-то почистил кэш
3. Собирается проект 2 - POISONED
4. Собирается проект 1 - POISONED



У общих CI-агентов
общий кэш!

Изоляция сборки

Изоляция сборки

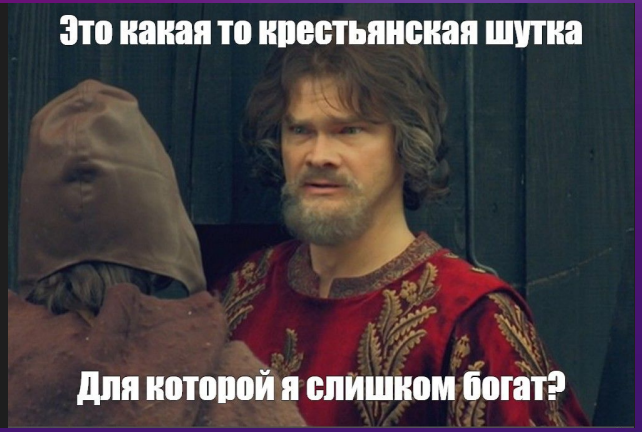
- Чистка кэша перед каждой сборкой

```
dotnet nuget locals clear --all

docker system prune -af

npm cache clean --force

yarn cache clean
```

Изоляция сборки

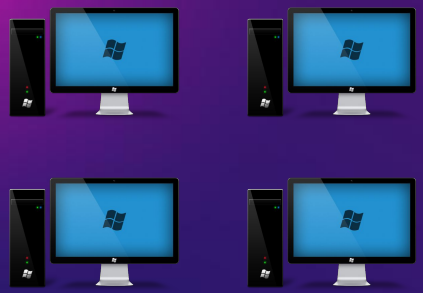
- Чистка кэша перед каждой сборкой

```
dotnet nuget locals clear --all

docker system prune -af

npm cache clean --force

yarn cache clean
```



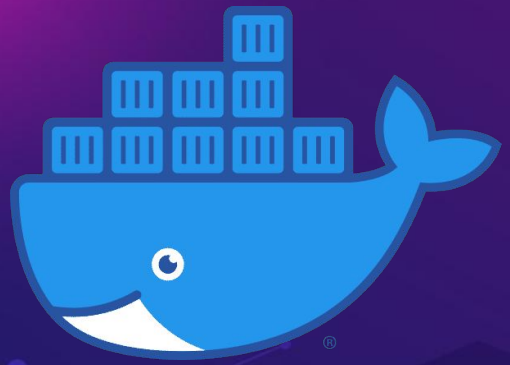
Изоляция сборки

- Чистка кэша перед каждой сборкой
- Раздельные агенты для разных команд, проектов, продуктов и т.д.



Изоляция сборки

- Чистка кэша перед каждой сборкой
- Раздельные агенты для разных команд, проектов, продуктов и т.д.
- Контейнеризация сборки



Ну теперь-то мы в безопасности?

14

Version range

Что такое version range?

Указание конкретных версий или диапазона версий для ваших зависимостей

npm

Value	Description
~ version	Only accept new patch versions
^ version	Only accept new minor and patch versions
version	Match exact version
> version	Must be greater than version
< version	Must be less than version
*	Matches any version

https://docs.npmjs.com/cli/v7/configuring-npm/package-json

npm

Value	Description
~ version	Only accept new patch versions
^ version	Only accept new minor and patch versions
version	Match exact version
> version	Must be greater than version
< version	Must be less than version
*	Matches any version

https://docs.npmjs.com/cli/v7/configuring-npm/package-json

npm

```
"dependencies": {
  "chalk": "^4.1.2",
  "commander": "^4.1.1",
  "cross-spawn": "^7.0.3",
  "envinfo": "^7.8.1",
  "fs-extra": "^10.0.0",
  "hyperquest": "^2.1.3",
  "prompts": "^2.4.2",
  "semver": "^7.3.5",
  "tar-pack": "^3.4.1",
  "tmp": "^0.2.1",
  "validate-npm-package-name": "^3.0.0"
},
```

https://github.com/facebook/create-react-app/blob/main/packages/create-react-app/package.json

Ну теперь-то мы в безопасности?

15

Lockfile

Что такое lockfile?

Это дополнительный файл, который генерируется автоматически и хранит в себе полное дерево всех зависимостей с версиями.

Lockfile

- Фиксация версии

Lockfile

- Фиксация версии
- Воспроизводимость сборки

Lockfile

- Фиксация версии
- Воспроизводимость сборки
- Контроль

Lockfile

- Фиксация версии
- Воспроизводимость сборки
- Контроль
- Безопасность

yarn.lock

```
# THIS IS AN AUTOGENERATED FILE. DO NOT EDIT THIS FILE DIRECTLY.
# yarn lockfile v1


"@babel/code-frame@7.10.4":
  version "7.10.4"
  resolved "https://npm.mycompany.com/npm/@babel/code-frame/-/code-frame-7.10.4.tgz#168da1a36e90da68ae8d49c0f1b48c7c624
  integrity sha512-vG6SvB6oYEhvgisZNFRmRCUkLz11c7rp+tbNTynGqc6mS1d5ATd/sGyV6W0KZZnXRKMTzZDRgQT3Ou9jhpAfUg==
  dependencies:
    "@babel/highlight" "^7.10.4"

"@babel/code-frame@7.12.11":
  version "7.12.11"
  resolved "https://npm.mycompany.com/npm/@babel/code-frame/-/code-frame-7.12.11.tgz#f4ad435aa263db935b8f10f2c552d23fb7
  integrity sha512-Zt1yodBx1UcyiePMSkWnU4hPqhwq7hGi2nFL1LeA3EUl+q2LQx16MISgJ0+z7dnmgvP9QtIleuETGOiOH1RcIw==
  dependencies:
    "@babel/highlight" "^7.10.4"
```

https://docs.npmjs.com/cli/v8/configuring-npm/package-lock-json

yarn.lock

```
# THIS IS AN AUTOGENERATED FILE. DO NOT EDIT THIS FILE DIRECTLY.
# yarn lockfile v1


"@babel/code-frame@7.10.4":
  version "7.10.4"
  resolved "https://npm.mycompany.com/npm/@babel/code-frame/-/code-frame-7.10.4.tgz#168da1a36e90da68ae8d49c0f1b48c7c624
  integrity sha512-vG6SvB6oYEhvgisZNFRmRCUkLz11c7rp+tbNTynGqc6mS1d5ATd/sGyV6W0KZZnXRKMTzZDRgQT3Ou9jhpAfUg==
  dependencies:
    "@babel/highlight" "^7.10.4"

"@babel/code-frame@7.12.11":
  version "7.12.11"
  resolved "https://npm.mycompany.com/npm/@babel/code-frame/-/code-frame-7.12.11.tgz#f4ad435aa263db935b8f10f2c552d23fb7
  integrity sha512-Zt1yodBx1UcyiePMSkWnU4hPqhwq7hGi2nFL1LeA3EUl+q2LQx16MISgJ0+z7dnmgvP9QtIleuETGOiOH1RcIw==
  dependencies:
    "@babel/highlight" "^7.10.4"
```

https://docs.npmjs.com/cli/v8/configuring-npm/package-lock-json

yarn.lock

```
# THIS IS AN AUTOGENERATED FILE. DO NOT EDIT THIS FILE DIRECTLY.
# yarn lockfile v1


"@babel/code-frame@7.10.4":
  version "7.10.4"
  resolved "https://npm.mycompany.com/npm/@babel/code-frame/-/code-frame-7.10.4.tgz#168da1a36e90da68ae8d49c0f1b48c7c624
  integrity sha512-vG6SvB6oYEhvgisZNFRmRCUkLz11c7rp+tbNTynGqc6mS1d5ATd/sGyV6W0KZZnXRKMTzZDRgQT3Ou9jhpAfUg==
  dependencies:
    "@babel/highlight" "^7.10.4"

"@babel/code-frame@7.12.11":
  version "7.12.11"
  resolved "https://npm.mycompany.com/npm/@babel/code-frame/-/code-frame-7.12.11.tgz#f4ad435aa263db935b8f10f2c552d23fb7
  integrity sha512-Zt1yodBx1UcyiePMSkWnU4hPqhwq7hGi2nFL1LeA3EUl+q2LQx16MISgJ0+z7dnmgvP9QtIleuETGOiOH1RcIw==
  dependencies:
    "@babel/highlight" "^7.10.4"
```

https://docs.npmjs.com/cli/v8/configuring-npm/package-lock-json

yarn.lock

```
# THIS IS AN AUTOGENERATED FILE. DO NOT EDIT THIS FILE DIRECTLY.
# yarn lockfile v1


"@babel/code-frame@7.10.4":
  version "7.10.4"
  resolved "https://npm.mycompany.com/npm/@babel/code-frame/-/code-frame-7.10.4.tgz#168da1a36e90da68ae8d49c0f1b48c7c624
  integrity sha512-vG6SvB6oYEhvgisZNFRmRCUkLz11c7rp+tbNTynGqc6mS1d5ATd/sGyV6W0KZZnXRKMTzZDRgQT3Ou9jhpAfUg==
  dependencies:
    "@babel/highlight" "^7.10.4"

"@babel/code-frame@7.12.11":
  version "7.12.11"
  resolved "https://npm.mycompany.com/npm/@babel/code-frame/-/code-frame-7.12.11.tgz#f4ad435aa263db935b8f10f2c552d23fb7
  integrity sha512-Zt1yodBx1UcyiePMSkWnU4hPqhwq7hGi2nFL1LeA3EUl+q2LQx16MISgJ0+z7dnmgvP9QtIleuETGOiOH1RcIw==
  dependencies:
    "@babel/highlight" "^7.10.4"
```

https://docs.npmjs.com/cli/v8/configuring-npm/package-lock-json

Nuget packages.lock.json


```
"version": 1,
  "dependencies": {
    ".NETCoreApp,Version=v5.0": {
      "JetBrains.Annotations": {
        "type": "Direct",
        "requested": "[2021.2.0, )",
        "resolved": "2021.2.0",
        "contentHash": "kKSyoVfndMriKHLfYGmr0uzQuI4jcc3TKGyww7buJFCYeHb/X0kodYBPL7n9454q7v6ASiRmDgpPGaDGerg/Hg=="
      },
      "Microsoft.Extensions.DependencyInjection": {
        "type": "Transitive",
        "resolved": "5.0.0",
        "contentHash": "Rc2kb/p3Ze6cP6rhFC3PJRdWGbLvSHZc0ev7YlyeU6FmHciDMLrhoVoTUEzKPhN5ZjFgKF1Cf5fOz8mCMIkvpA==",
        "dependencies": {
          "Microsoft.Extensions.DependencyInjection.Abstractions": "5.0.0"
        }
      }
```


https://docs.microsoft.com/en-us/nuget/consume-packages/package-references-in-project-files

Nuget packages.lock.json


```
"version": 1,
  "dependencies": {
    ".NETCoreApp,Version=v5.0": {
      "JetBrains.Annotations": {
        "type": "Direct",
        "requested": "[2021.2.0, )",
        "resolved": "2021.2.0",
        "contentHash": "kKSyoVfndMriKHLfYGmr0uzQuI4jcc3TKGyww7buJFCYeHb/X0kodYBPL7n9454q7v6ASiRmDgpPGaDGerg/Hg=="
      },
      "Microsoft.Extensions.DependencyInjection": {
        "type": "Transitive",
        "resolved": "5.0.0",
        "contentHash": "Rc2kb/p3Ze6cP6rhFC3PJRdWGbLvSHZc0ev7YlyeU6FmHciDMLrhoVoTUEzKPhN5ZjFgKF1Cf5fOz8mCMIkvpA==",
        "dependencies": {
          "Microsoft.Extensions.DependencyInjection.Abstractions": "5.0.0"
        }
      }
```


https://docs.microsoft.com/en-us/nuget/consume-packages/package-references-in-project-files

Фиксируй!

1. Важно контролировать версии ваших зависимостей

Фиксируй!

1. Важно контролировать версии ваших зависимостей
2. Не используйте диапазоны в версионировании

Фиксируй!

1. Важно контролировать версии ваших зависимостей
2. Не используйте диапазоны в версионировании
3. Транзитивные зависимости могут использовать диапазон версий в своих зависимостях


Best
Practices

Best Practices

1. Используйте команды для проверки deprecated и vulnerable версий пакетов

```
dotnet list package --deprecated

dotnet list package --vulnerable

npm audit
```

Best Practices

1. Используйте команды для проверки deprecated и vulnerable версий пакетов

```
C:\Users\Jon-Personal\ContosoVulnerability>dotnet list package --vulnerable --include-transitive

The following sources were used:
   https://api.nuget.org/v3/index.json
   C:\Program Files (x86)\Microsoft SDKs\NuGetPackages\

Project `ContosoVulnerability` has the following vulnerable packages
   [net5.0]:
   Top-level Package            Requested   Resolved   Severity   Advisory URL
   > Auth0-WCF-Service-JWT      1.0.3       1.0.3      Critical   https://github.com/advisories/GHSA-qpvx-gpqm-g98j
   > UmbracoForms               8.4.1       8.4.1      Moderate   https://github.com/advisories/GHSA-8m73-w2r2-6xxj

   Transitive Package          Resolved   Severity   Advisory URL
   > Microsoft.Data.OData      5.2.0      Moderate   https://github.com/advisories/GHSA-mv2r-q4g5-j8q5
```

Best Practices

1. Используйте команды для проверки deprecated и vulnerable версий пакетов
2. Блокировка прямого доступа к nuget.org или registry.npmjs.org

Best Practices

1. Используйте команды для проверки deprecated и vulnerable версий пакетов
2. Блокировка прямого доступа к nuget.org или registry.npmjs.org
3. Полный отказ от nuget.org или npmjs.org

А что мне делать сейчас?

А что мне делать сейчас?

1. Изучайте инструменты

А что мне делать сейчас?

1. Изучайте инструменты
2. Реализуйте общие рекомендации (lock file, config file, изолируйте сборки)

А что мне делать сейчас?

1. Изучайте инструменты
2. Реализуйте общие рекомендации (lock file, config file, изолируйте сборки)
3. Следите за уязвимостями в активно используемых компонентах или библиотеках в компании

А что мне делать сейчас?

1. Изучайте инструменты
2. Реализуйте общие рекомендации (lock file, config file, изолируйте сборки)
3. Следите за уязвимостями в активно используемых компонентах или библиотеках в компании
4. Своевременно обновляйтесь в случае обнаружения уязвимостей, но крайне осторожно!

Полезно знать

- Building a supply chain attack with .NET, NuGet, DNS, source generators, and more
- 3 Ways to Mitigate Risk When Using Private Package Feeds
- Supply chain attack
- Best practices for a secure software supply chain
- Dependency Confusion: How I Hacked Into Apple, Microsoft and Dozens of Other Companies
- Introducing SLSA, an End-to-End Framework for Supply Chain Integrity

Вопросы?

CREDITS

CREDITS: This presentation template was created by **Slidesgo,** and includes icons by **Flaticon**, and infographics & images by **Freepik**