

БЕЗОПАСНОСТЬ SUPPLY CHAIN ВАШИХ OPEN SOURCE- ЗАВИСИМОСТЕЙ

ТАТЬЯНА КУЦОВОЛ
ВЕДУЩИЙ АНАЛИТИК-ИССЛЕДОВАТЕЛЬ ИБ
ГК «СОЛАР»

О спикере



@luttatiana



t.kutsovol@rt-solar.ru



ТАТЬЯНА КУЦОВОЛ

ВЕДУЩИЙ АНАЛИТИК-ИССЛЕДОВАТЕЛЬ ИБ,
ГК «СОЛАР»

План-капкан

[01]

Общее понятие безопасности цепочки поставок (Supply Chain) в контексте разработки

[02]

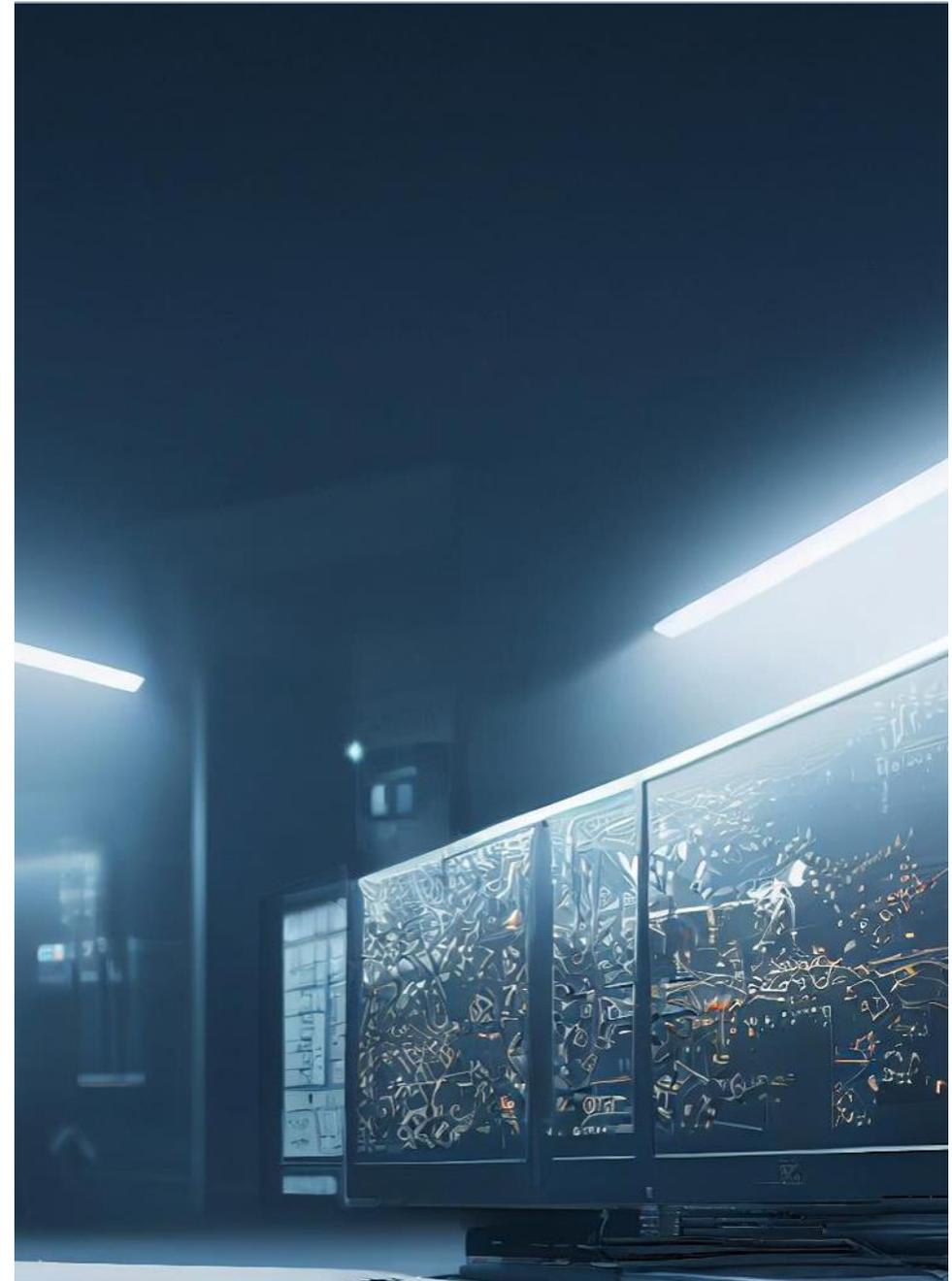
Обзор мировых стандартов SLSA, OWASP SCVS, CIS SSCSG

[03]

Разбор атак на цепочку поставок (Supply Chain).
Примеры техник злоумышленников и подходы детектирования

[04]

Оценка риска компрометации через Supply Chain зависимостей. Детальный обзор метрик для сторонних компонент, которые можно и нужно смотреть



Вводные

- Open Source-зависимости **везде**
- **НЕ использовать** Open Source-зависимости **невозможно**
- **Использовать** Open Source-зависимости **важно безопасно**

SECURE SUPPLY CHAIN

Вводные

“You can't trust code that you did not totally create yourself. Especially code from companies that employ people like me”.

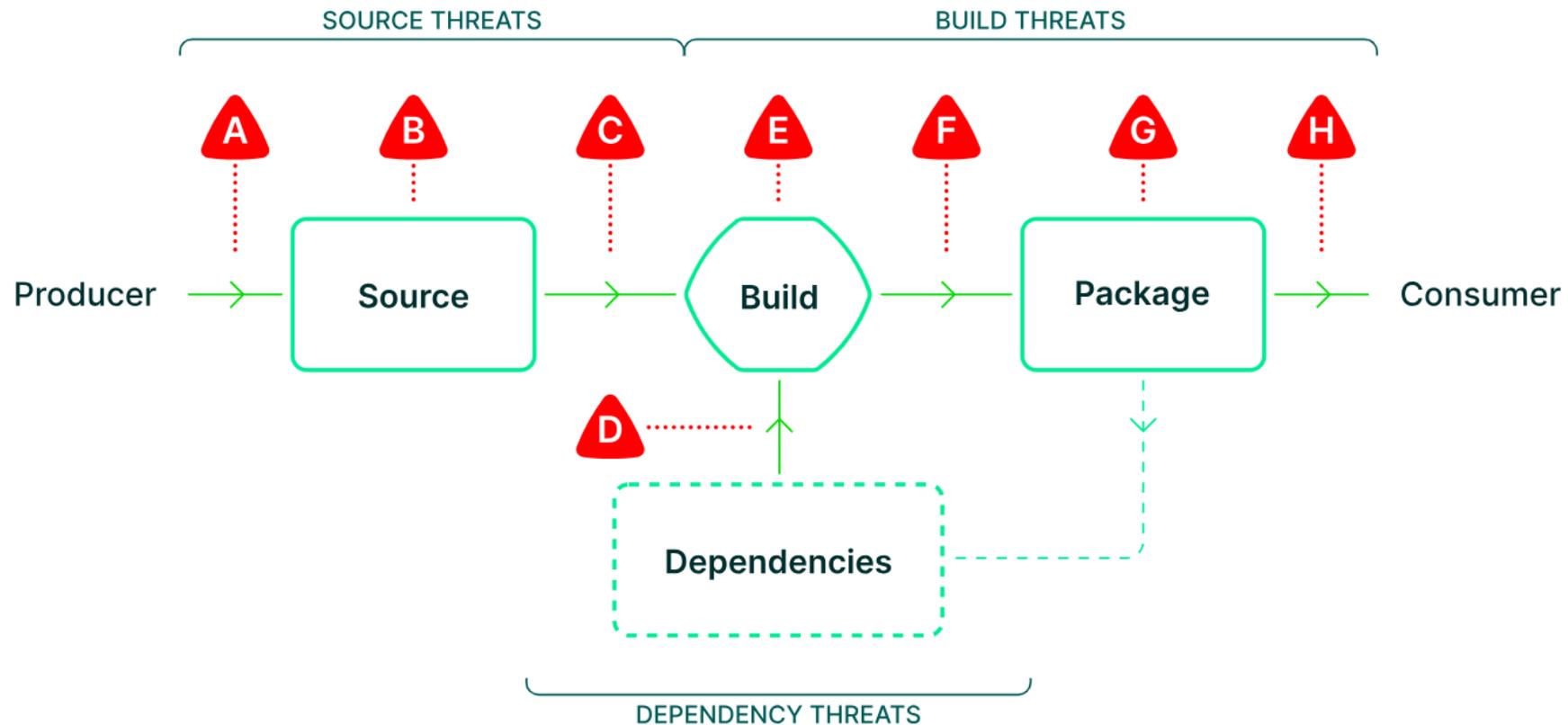
Вы можете доверять коду, только если вы написали его сами. Особенно нельзя доверять коду от компаний, которые берут на работу таких как я”.



Ken Thompson, 1984
“Reflections on Trusting Trust”

Безопасность цепочки поставок (Supply Chain)

Обеспечение безопасности **на всех этапах пути**, по которому ПО попадает в организацию, от момента его создания или покупки до этапа использования.





ПОЧЕМУ БЕЗОПАСНОСТЬ SUPPLY CHAIN – ПРОБЛЕМА?

Почему безопасность Supply Chain – проблема?

event-stream в 2018 г. -
завладели популярной
библиотекой в NPM

Packt Hu

Malicious code in npm 'event-stream' package targets a bitcoin wallet and causes 8 million download...

Last week Ayrton Sparling, a Computer Science major at CSUF, California disclosed that the popular npm package, event-stream, contains a...

28 нояб. 2018 г.



SolarWinds в 2020 г. -
поставка уязвимых
обновлений для
клиентов

В Ведомости

Reuters: при атаке на SolarWinds хакеры украли данные контрразведки США

Хакеры, взломавшие IT-компанию SolarWinds в прошлом году, похитили данные о контрразведывательных операциях США и санкционной политике...

7 окт. 2021 г.



VMConnect в 2023 г. -
вредоносный пакет в
PyPI, похожий на
vConnector

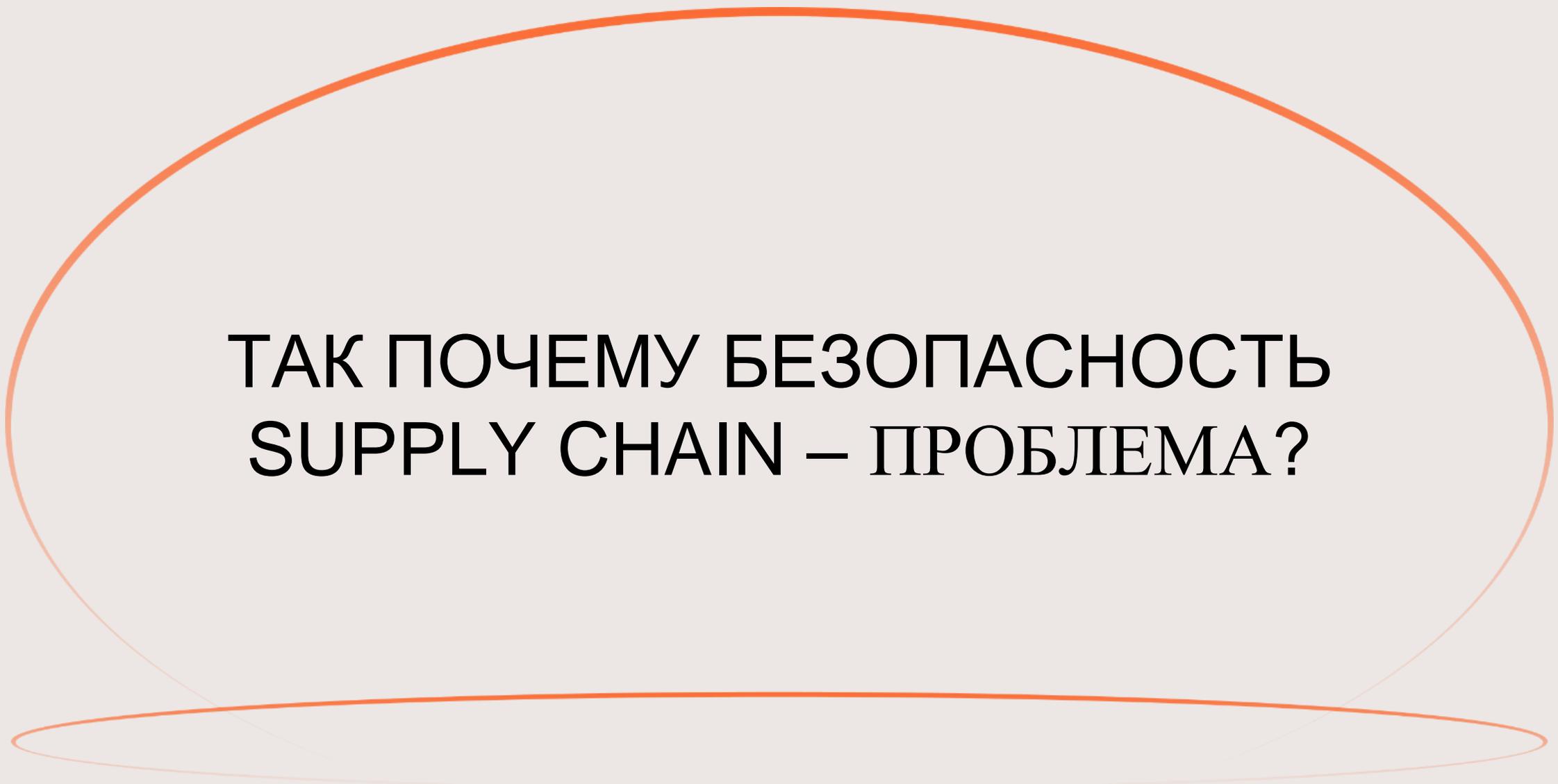
Bleeping Computer

Lazarus hackers deploy fake VMware PyPI packages in VMConnect attacks

North Korean state-sponsored hackers have uploaded malicious packages to the PyPI (Python Package Index) repository, camouflaging one of...

31 авг. 2023 г.





**ТАК ПОЧЕМУ БЕЗОПАСНОСТЬ
SUPPLY CHAIN – ПРОБЛЕМА?**

Почему безопасность Supply Chain – проблема?

на **742%**

в год в среднем растет количество атак
через цепочку поставок





НО ПОЧЕМУ ВСЕ ЖЕ
БЕЗОПАСНОСТЬ
SUPPLY CHAIN – ПРОБЛЕМА?

Почему безопасность Supply Chain – проблема?



MAY 12, 2021

Executive Order on Improving the Nation's Cybersecurity



[BRIEFING ROOM](#)

[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

Share



Почему безопасность Supply Chain – проблема?



commitments may include state... requirements to complete a vendor's current stage, next steps, and points of contact for questions;

- (iii) incorporating automation throughout the lifecycle of FedRAMP, including assessment, authorization, continuous monitoring, and compliance;
- (iv) digitizing and streamlining documentation that vendors are required to complete, including through online accessibility and pre-populated forms; and
- (v) identifying relevant compliance frameworks, mapping those frameworks onto requirements in the FedRAMP authorization process, and allowing those frameworks to be used as a substitute for the relevant portion of the authorization process, as appropriate.

Sec. 4. Enhancing Software Supply Chain Security.

(a) The security of software used by the Federal Government is vital to the Federal Government's ability to perform its critical functions. The development of commercial software often lacks the rigor, sufficient focus on the ability of the software to resist attacks, and adequate controls to prevent tampering by malicious actors. There is a need to implement more rigorous and predictable mechanisms for ensuring that products function securely, and as intended. The security and integrity of "critical software" – software that performs functions critical to trust (such as affording or requiring elevated system privileges or direct access to networking and computing resources) – is a particular concern. Accordingly, the Federal Government must take action to rapidly improve the security and integrity of the software supply chain, with a priority on addressing critical software.



Share



Почему безопасность Supply Chain – проблема?



commitments may include state-of-the-art requirements to complete a vendor's current stage, next steps, and points of contact for questions;
(iii) incorporating automation throughout the lifecycle of FedRAMP, including assessment, authorization, continuous monitoring, and compliance;

Sec. 4. Enhancing Software Supply Chain Security.

(a) The security of software used by the Federal Government is

of the authorization process, as appropriate.

Sec. 4. Enhancing Software Supply Chain Security.

(a) The security of software used by the Federal Government is vital to the Federal Government's ability to perform its critical functions. The development of commercial software often lacks the security, sufficient focus on the ability of the software to resist attacks, and adequate controls to prevent tampering by malicious actors. There is a need to implement more rigorous and predictable mechanisms for ensuring that products function securely, and as intended. The security and integrity of "critical software" – software that performs functions critical to trust (such as affording or requiring elevated system privileges or direct access to networking and computing resources) – is a particular concern. Accordingly, the Federal Government must take action to rapidly improve the security and integrity of the software supply chain, with a priority on addressing critical software.

Share



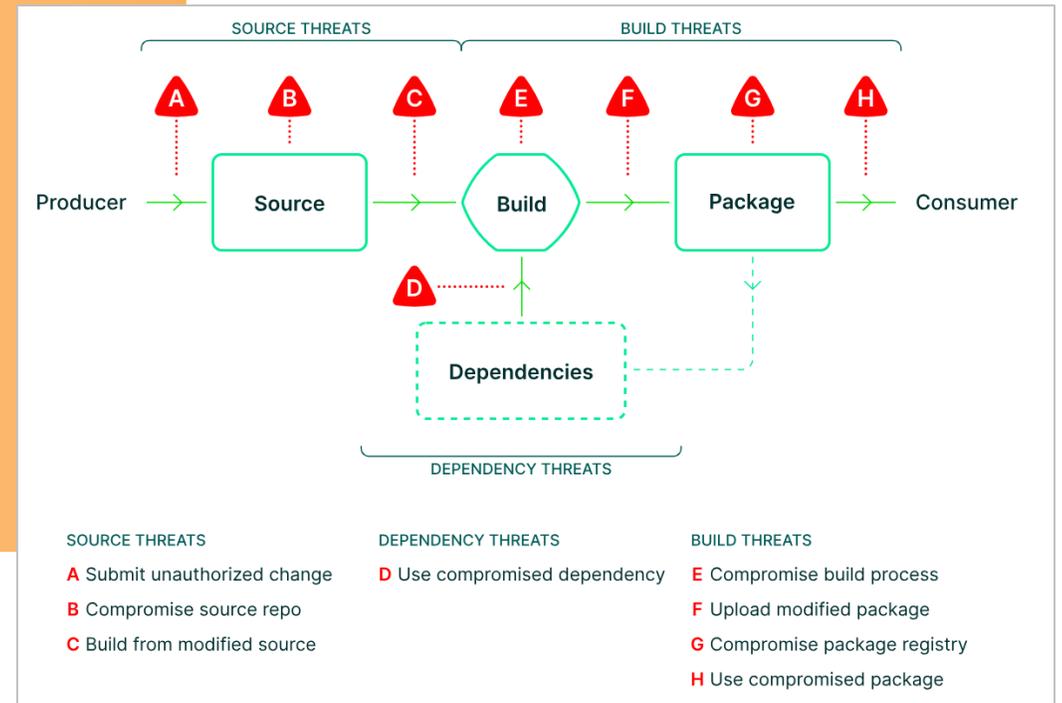
Обзор стандартов

- SLSA (Supply-chain Levels for Software Artifacts)
- OWASP Software Component Verification Standard (SCVS)
- CIS Software Supply Chain Security Guide (SSCSG)

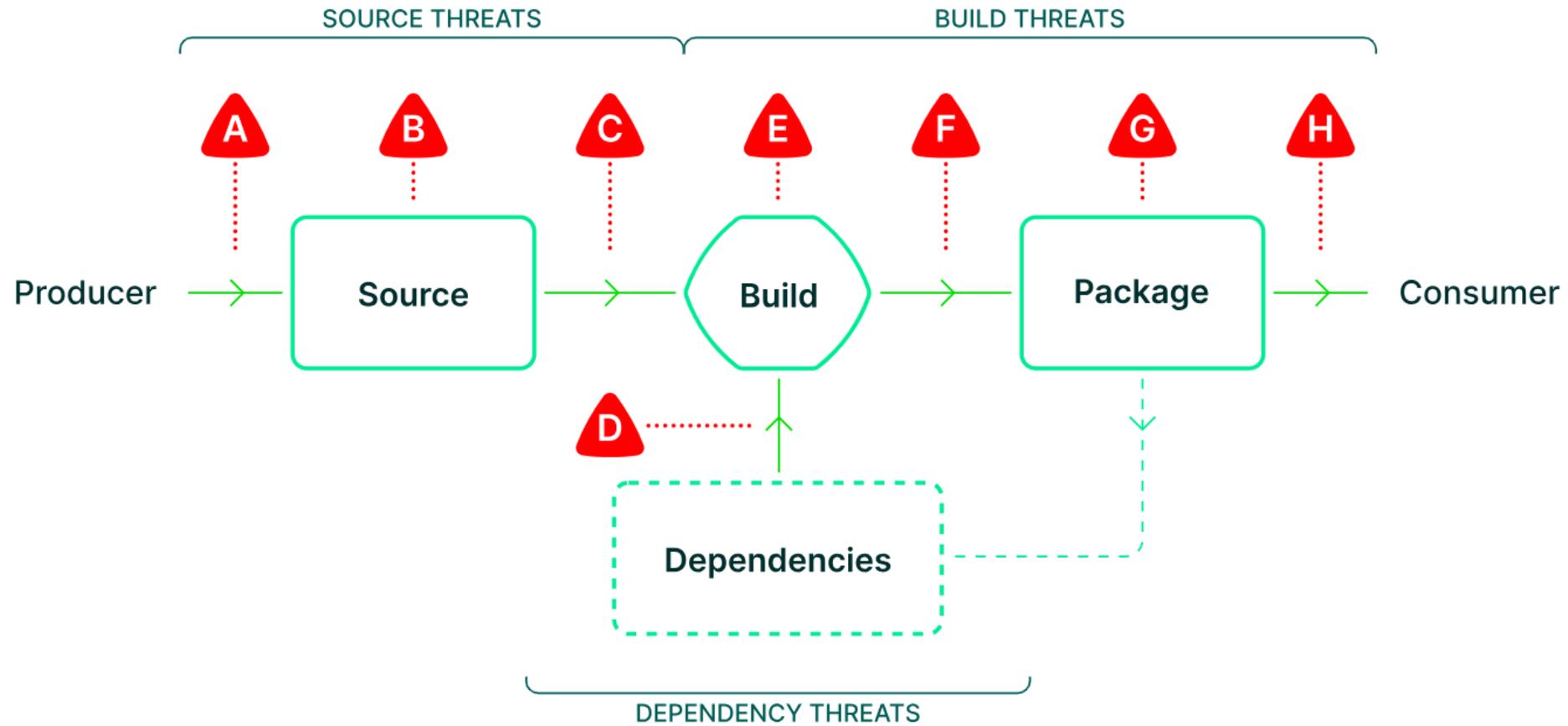
SLSA (Supply-chain Levels for Software Artifacts)

Фреймворк, чек-лист требований, которые должны быть учтены в безопасной цепочке поставок (Supply Chain)

SLSA
Draft Version 1.0
Open for Comments



SLSA (Supply-chain Levels for Software Artifacts)



SLSA (Supply-chain Levels for Software Artifacts)

Provenance – информация о том, кто создал один или несколько программных артефактов и какие этапы и материалы использовались для создания этих артефактов. Формат **in-toto** (<https://github.com/in-toto/attestation>).



SLSA (Supply-chain Levels for Software Artifacts)



Provenance – информация о том, кто создал один или несколько программных артефактов и какие этапы и материалы использовались для создания этих артефактов. Формат **in-toto** (<https://github.com/in-toto/attestation>).

```
{
  "_type": "https://in-toto.io/Statement/v0.1",
  "subject": [
    {
      "name": "output.txt",
      "digest": {
        "sha256": "a948904f2f0f479b8f8197694b30184b0d2ed1c1cd2a1ec0fb85d299a192a447"
      }
    }
  ],
  "predicateType": "https://slsa.dev/provenance/v0.2",
  "predicate": {
    "buildtype": "https://www.jenkins.io/Pipeline",
    "builder": {
      "id": "http://jenkisDashboard/job/public_test/122/"
    },
    "invocation": {
      "configSource": {
        "uri": "git://github.com/Samsung/slsa-jenkins-generator.git@refs/heads/main",

```

SLSA (Supply-chain Levels for Software Artifacts)

Gitlab платная версия - Опция `RUNNER_GENERATE_ARTIFACTS_METADATA = "true"` в вашем `.gitlab-ci.yml` файле



Why GitLab Platform Solutions Pricing Resources Company Contact us

Jun 22, 2022 - Dov Hershkovitch

GitLab 15.1 Release

GitLab 15.1 released with SAML Group Sync and SLSA level 2 build artifact attestation

Today, we are excited to announce the release of GitLab 15.1 with [SAML Group Sync](#), [SLSA level 2 build artifact attestation](#), [links to included CI/CD configuration](#), [enhanced visibility into value stream with DORA metrics](#), and much more!

These are just a few highlights from the 30+ improvements in this release. Read on to check out all of the great updates below.

Join us on June 23rd as we celebrate DevOps! GitLab co-founder and CEO, Sid Sijbrandij, will introduce best-selling author and DORA co-founder, Gene Kim. Gene will share his research and expectations for the future of DevOps then GitLab VP of Product, David DeSanto, will share how GitLab is evolving The One DevOps Platform to meet that future. We'll also unveil a new program to support your career aspirations. You won't want to miss this one-hour virtual event. [Reserve your seat](#) now!

To preview what's coming in next month's release, check out our [Upcoming Releases page](#), which includes our 15.2 release kickoff video.

Get free trial Sign in

SLSA (Supply-chain Levels for Software Artifacts)

GitHub Actions с плагином - <https://github.com/slsa-framework/slsa-github-generator>
Для Jenkins - <https://github.com/slsa-framework/slsa-jenkins-generator>

The image shows two overlapping screenshots of GitHub repository pages. The background screenshot is for the repository `slsa-framework / slsa-github-generator`. It shows the repository name, a search bar, navigation tabs for Code, Issues (237), Pull requests (10), Actions, and Projects, and statistics for Watch (11), Fork (106), and Star (334). A file browser on the left lists directories like `.github`, `actions`, `github`, `images`, `internal`, `signing`, `slsa`, `version`, and files like `.gitignore`, `.golangci.yml`, `.markdownlint.yml`, `.markdownlintignore`, and `.yamllint.yml`.

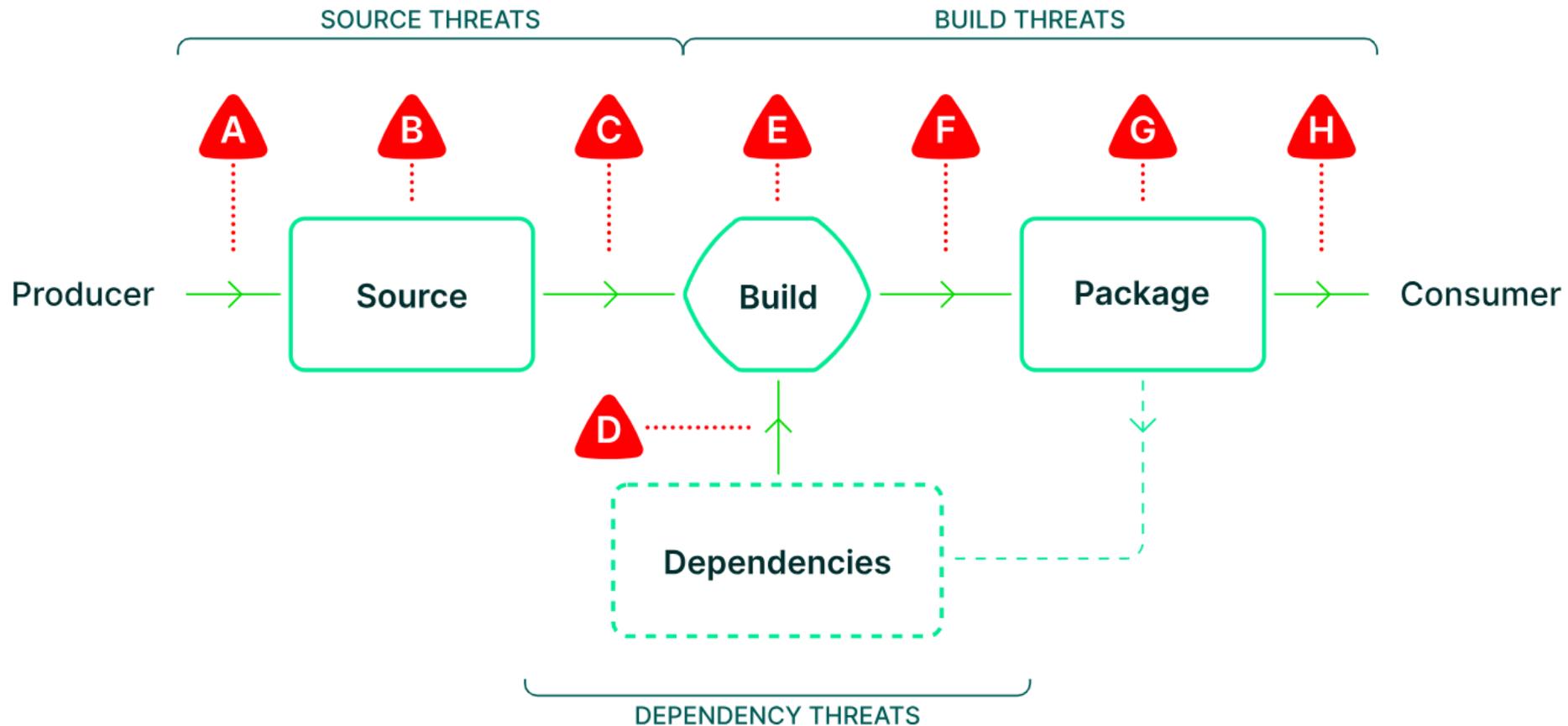
The foreground screenshot is for the repository `slsa-framework / slsa-jenkins-generator`. It shows the repository name, a search bar, navigation tabs for Code, Issues (1), Pull requests (2), Actions, Security, and Insights, and statistics for Watch (8), Fork (5), and Star (19). A commit history table is visible:

Commit	Message	Time
0e56af0	last year	12 Commits
docker	[DOC] Update License location	last year
example	Add example and config files	2 years ago
plugin	[DOC] Update License location	last year
LICENSE	[DOC] Update License location	last year
README.md	[DOC] Update README.md	last year
renovate.json	Add renovate.json (#1)	last year

On the right side of the foreground screenshot, there is an 'About' section with the following text: 'A proof-of-concept SLSA provenance generator for Jenkins'. Below this are links for Readme, MIT license, Activity, Custom properties, 19 stars, 8 watching, 5 forks, and a 'Report repository' link.

SLSA (Supply-chain Levels for Software Artifacts)

Нужен в первую очередь для производителя ПО, чтобы можно было сказать: «А вот у меня SLSA3, и вот мой provenance». Это дает больше доверия для пользователя и заставляет производителя ПО относиться к выбору сборки более тщательно.



SLSA (Supply-chain Levels for Software Artifacts)

Kyverno прошел third-party аудит, проводимый компанией Ada Logics, по итогу которого были также оценены риски цепочки поставок в соответствии с SLSA.

target one attack surface identified during the threat modelling. Policy bypasses, i.e. where an internal attacker attempts to submit a request that bypasses a policy deployed by the Kyverno admin.

The SLSA review found that Kyverno complies at the highest level (SLSA Level 3). Kyverno builds its releases on GitHub Actions and includes verifiable provenance with releases, which makes Kyverno hardened against a series of well-known attack vectors in Kyverno's software supply-chain.

3

Ada Logics Ltd



OWASP Software Component Verification Standard (SCVS)

Стандарт **OWASP SCVS**:

- направлен на улучшение безопасности и качества цепочек поставок программного обеспечения
- сформулирован достаточно кратко
- состоит из **шести контролей**



OWASP Software Component Verification Standard (SCVS)

V1: Необходимость инвентаризации

V2: Требования к содержимому спецификации программного обеспечения (SBOM)

V3: Требования к правильной настройке среды сборки

V4: Требования к управлению пакетами

V5: Анализ компонентов на уязвимости, качество и наличие опасных лицензий

V6: Сбор информации о происхождении пакетов и их модификаций



OWASP Software Component Verification Standard (SCVS)



OWASP SCVS [Get Started](#) [BOM Maturity Model](#)   

OWASP SCVS V1.0

- Frontispiece
- Preface
- Using the SCVS
- Assessment and Certification
- V1: Inventory Requirements
- V2: Software Bill of Materials (SBOM) Requirements
- V3: Build Environment Requirements
- V4: Package Management Requirements
- V5: Component Analysis Requirements
- V6: Pedigree and Provenance Requirements
- Guidance: Open Source Policy
- Appendix A: Glossary
- Appendix B: References

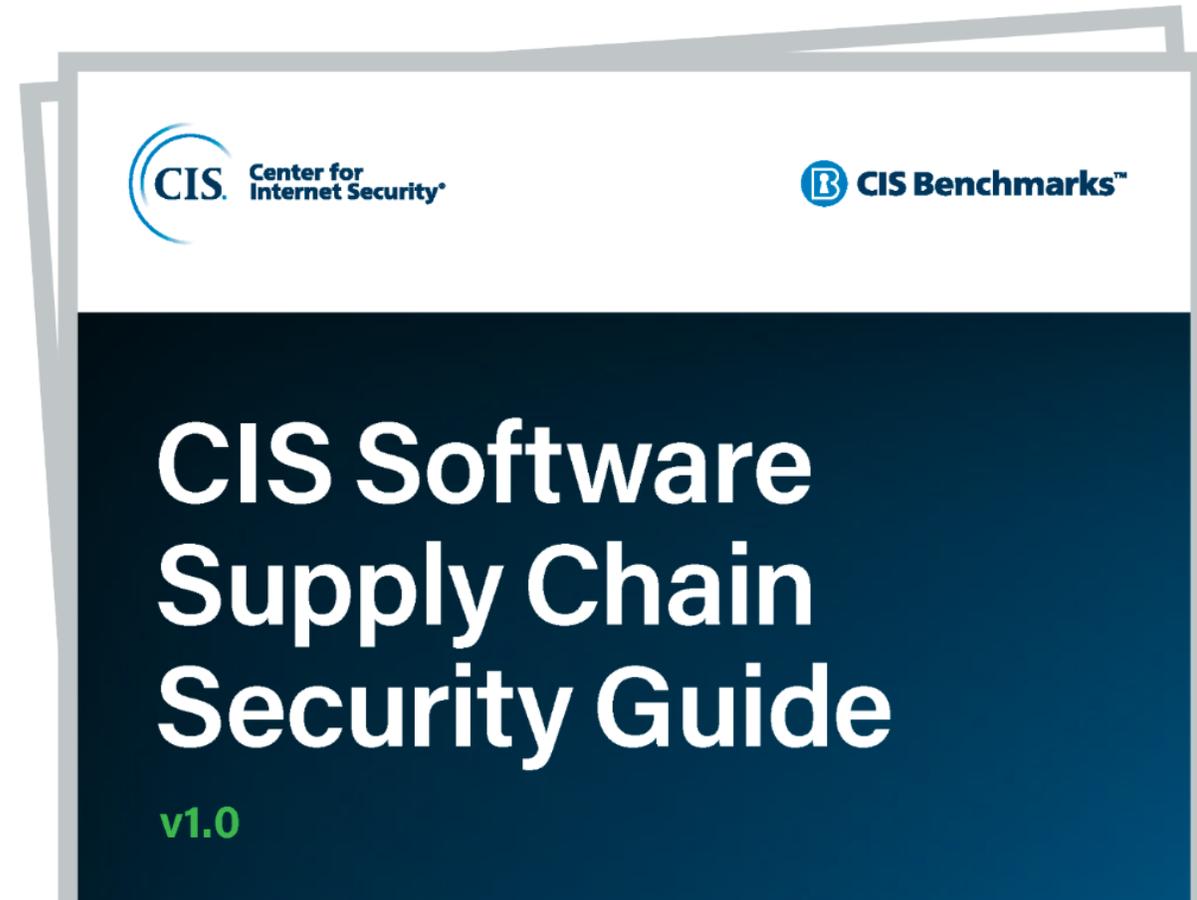
Verification Requirements

#	Description	L1	L2	L3
5.1	Component can be analyzed with linters and/or static analysis tools	✓	✓	✓
5.2	Component is analyzed using linters and/or static analysis tools prior to use		✓	✓
5.3	Linting and/or static analysis is performed with every upgrade of a component		✓	✓
5.4	An automated process of identifying all publicly disclosed vulnerabilities in third-party and open source components is used	✓	✓	✓
5.5	An automated process of identifying confirmed dataflow exploitability is used			✓
5.6	An automated process of identifying non-specified component versions is used	✓	✓	✓
5.7	An automated process of identifying out-of-date components is used	✓	✓	✓
5.8	An automated process of identifying end-of-life / end-of-support components is used			✓
5.9	An automated process of identifying component type is used		✓	✓
5.10	An automated process of identifying component function is used			✓
5.11	An automated process of identifying component quantity is used	✓	✓	✓
5.12	An automated process of identifying component license is used	✓	✓	✓

CIS Software Supply Chain Security Guide (SSCSG)

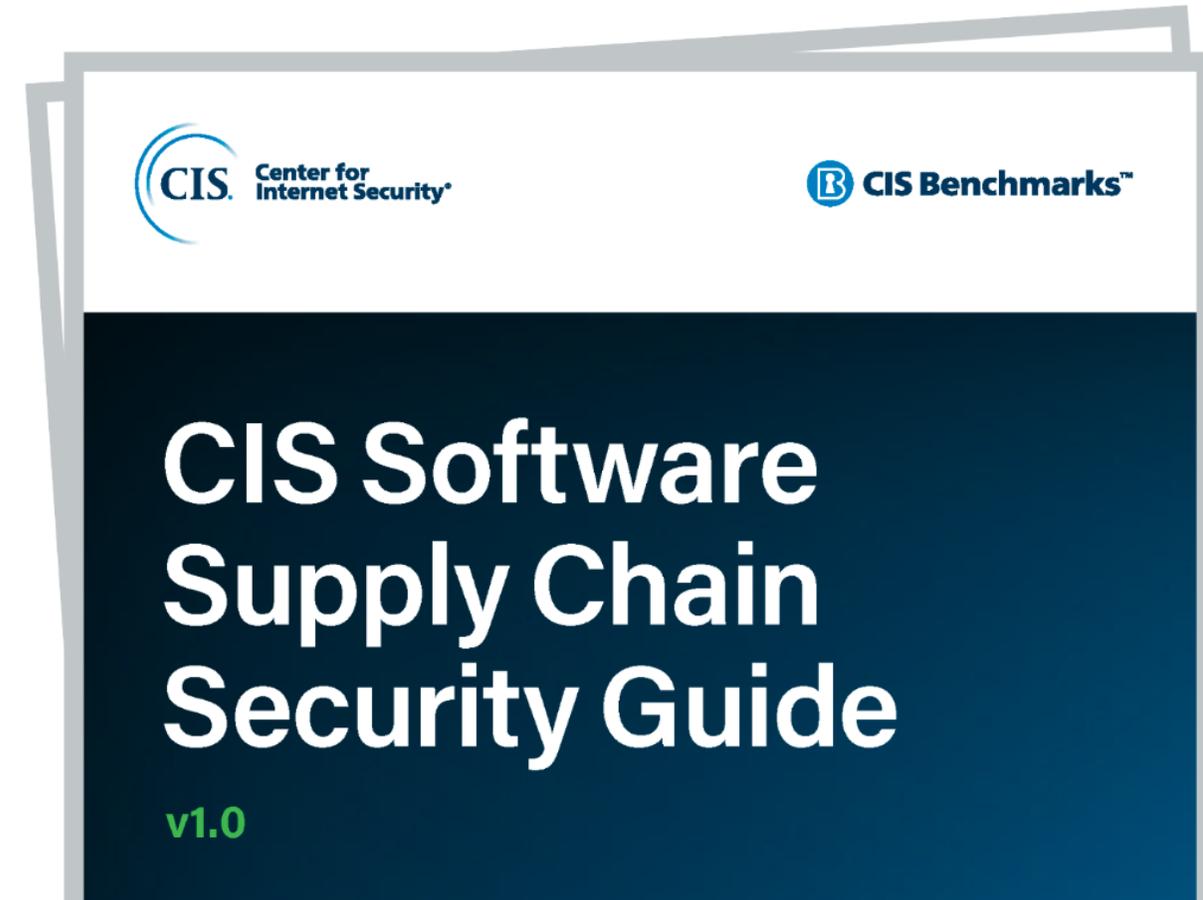
Гайд описывает этапы цепочки поставок программного обеспечения, начиная с добавления кода разработчиком и заканчивая доставкой ПО клиенту.

Руководство согласуется со стандартами безопасности, такими как SLSA, и включает более **100 рекомендаций в пяти основных категориях**.



CIS Software Supply Chain Security Guide (SSCSG)

1. **Source Code:** управление исходным кодом приложения
2. **Build Pipelines:** безопасность сборки
3. **Dependencies:** управление зависимостями
4. **Artifacts:** управление артефактами сборки
5. **Deployment:** защита процесса развертывания приложения и связанных файлов и конфигураций



CIS Software Supply Chain Security Guide (SSCSG)

ПРИМЕР

Согласно пункту **3.1.8 CIS Software Supply Chain Security Guidelines**, возраст пакета должен быть больше 60 дней: эта мера предосторожности поможет избежать внедрения потенциально вредоносных сторонних пакетов и обеспечить достаточное время на их проверку.

3.1.8 Ensure all packages used are more than 60 days old

Description

Use packages that are more than 60 days old.



Rationale

Third-party packages are a major risk since an organization cannot control them and there is always the possibility these packages could be malicious. It is a best practice to remain cautious with any third-party or open-source packages, until they can be verified that they are safe to use. Avoiding a new package gives an organization time to fully examine it, its maintainer, and its behavior, and gives the organization time to determine whether or not to use it.

NOTE Developers may not use packages that are less than 60 days old.

Audit

For every package used, ensure it is more than 60 days old.

Remediation

If a package used is less than 60 days old, stop using it and find another package.

Обзор стандартов

- SLSA (Supply-chain Levels for Software Artifacts)
- OWASP Software Component Verification Standard (SCVS)
- CIS Software Supply Chain Security Guide (SSCSG)

SLSA
Draft Version 1.0
Open for Comments



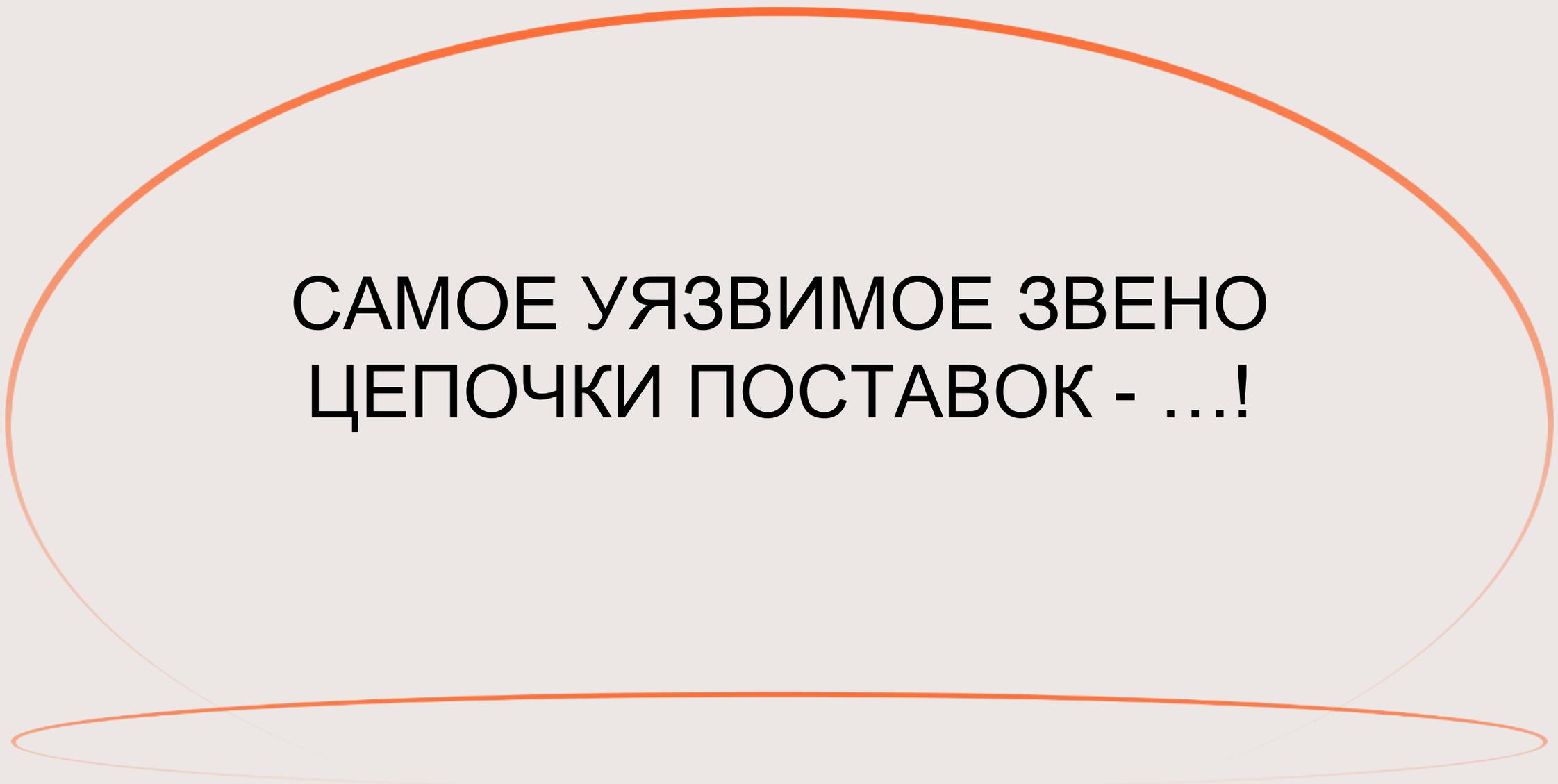
 **OWASP** | Standard
Open Web Application
Security Project

SCVS
Software Component
Verification Standard
Version 1.0

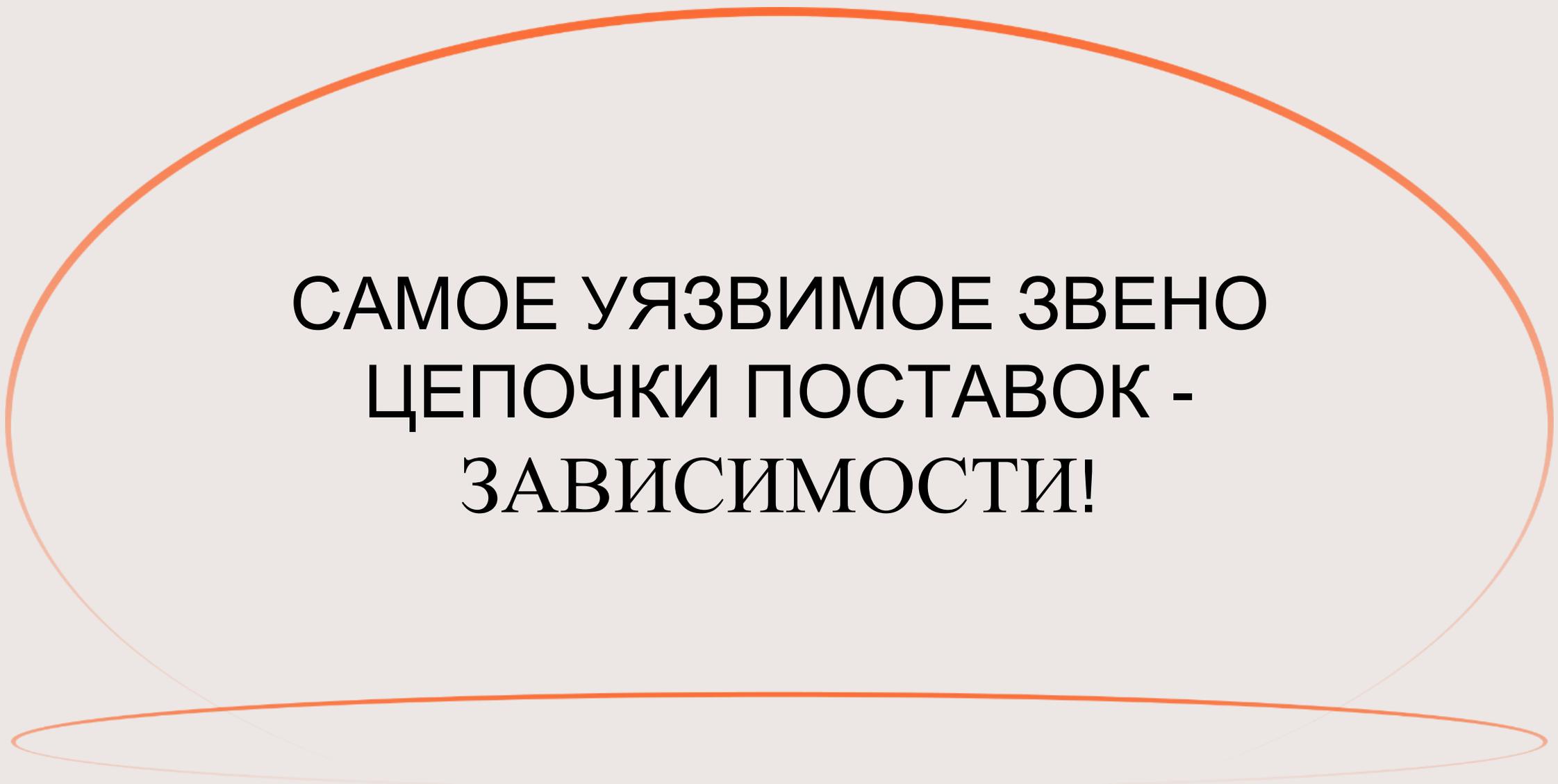
 **CIS** Center for
Internet Security™

 **CIS Benchmarks™**

**CIS Software
Supply Chain
Security Guide**



САМОЕ УЯЗВИМОЕ ЗВЕНО
ЦЕПОЧКИ ПОСТАВОК - ...!



**САМОЕ УЯЗВИМОЕ ЗВЕНО
ЦЕПОЧКИ ПОСТАВОК -
ЗАВИСИМОСТИ!**

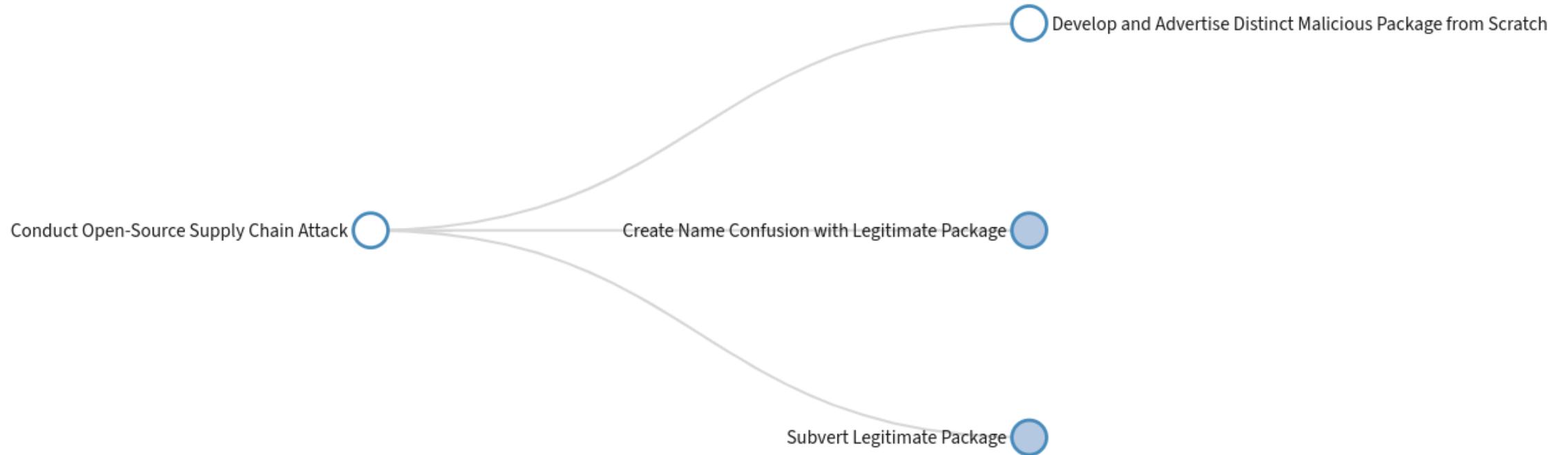
Вспомним

“You can't trust code that you did not totally create yourself. Especially code from companies that employ people like me”.

Вы можете доверять коду, только если вы написали его сами. Особенно нельзя доверять коду от компаний, которые берут на работу таких как я”.

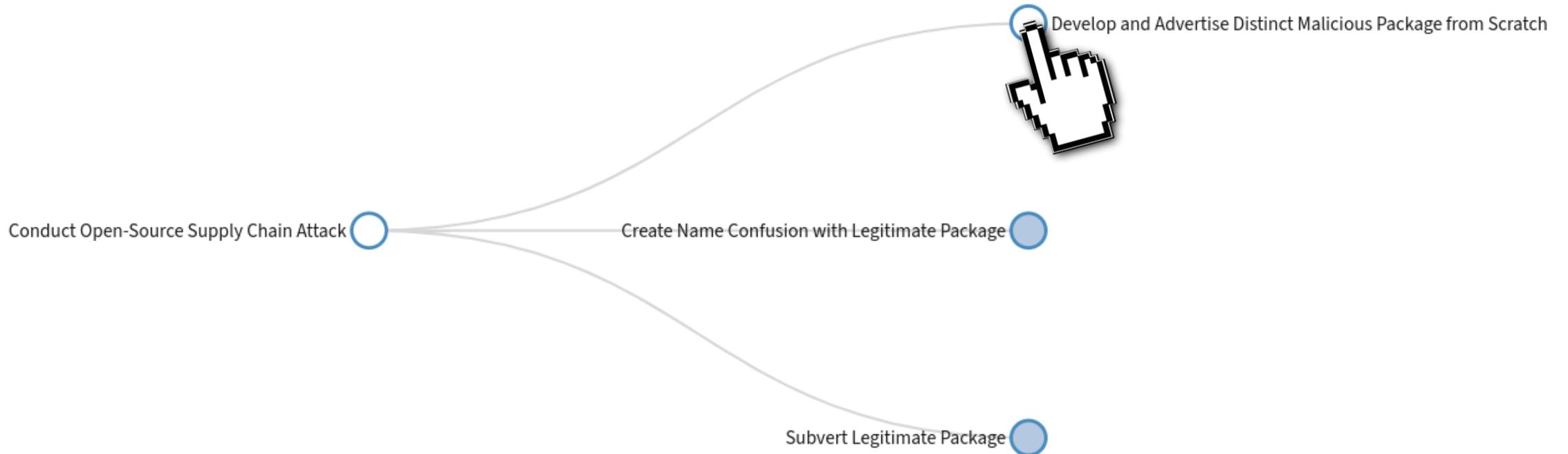
Ken Thompson, 1984
“Reflections on Trusting Trust”

Разбор атак на цепочку поставок (Supply Chain)



Разбор атак на цепочку поставок (Supply Chain)

Develop and Advertise Distinct Malicious Package from Scratch
(Разработка и реклама отдельного вредоносного пакета с нуля)



Разбор атак на цепочку поставок (Supply Chain)

Разработчик RIAEvangelist



Разбор атак на цепочку поставок (Supply Chain)

Разработчик RIAEvangelist



SC Media
<https://www.scmagazine.com> > ...

What happens when 'protestware' sabotages open source ...

17 мар. 2022 г. — As **RIAEvangelist** updated Node-ipc, he updated the version numbers as well, triggering automatic updating of code for many downstream users. " ...

Student Pocket Guide
<https://www.thestudentpocketguide.com> > ...

RIAEvangelist's anti-war "protestware" bashed by FOSS ...

Hactivist **RIAEvangelist** sparked outrage beyond the FOSS (Free and Open Source Software) community after uploading "protestware" (read malware) to his own ...



dev.by
<https://devby.io> > Новости

Популярный npm-пакет удаляет и перезаписывает ...

18 мар. 2022 г. — Однако теперь обнаружилось, что некоторые версии известной библиотеки node-ipc, тоже поддерживаемой **RIAEvangelist**, содержат гораздо более ...

PortSwigger
<https://portswigger.net> > npm...

NPM maintainer targets Russian users with data-wiping ' ...

21 мар. 2022 г. — '**RIAEvangelist**' (aka Brandon Nozaki Miller) embedded malware – or 'protestware', as he dubbed it – into Node.JS module node-ipc's latest ...

Подобных хактивистов можно отслеживать в репозитории <https://github.com/toxic-repos/toxic-repos>

Разбор атак на цепочку поставок (Supply Chain)

Проверка авторского состава. Владелец **easy-stack** является тот самый хактивист RIAEvangelist, о котором мы говорили выше.

Скриншот интерфейса Solar appScreener. В центре экрана отображены результаты сканирования цепочки поставок (SCA) для файла JS_sbom.json. В левом меню видны пункты: Обзор, Подробные результаты, Сканирования, Экспорт отчёта, Сравнение сканирований, Настройки. В верхней части экрана — панель навигации с ссылками: Домашняя страница, Проекты, Группы проектов, Правила и наборы, Администрирование, О продукте. В правом верхнем углу — дата сканирования: 9/9 24.11.2023 17:16:22.

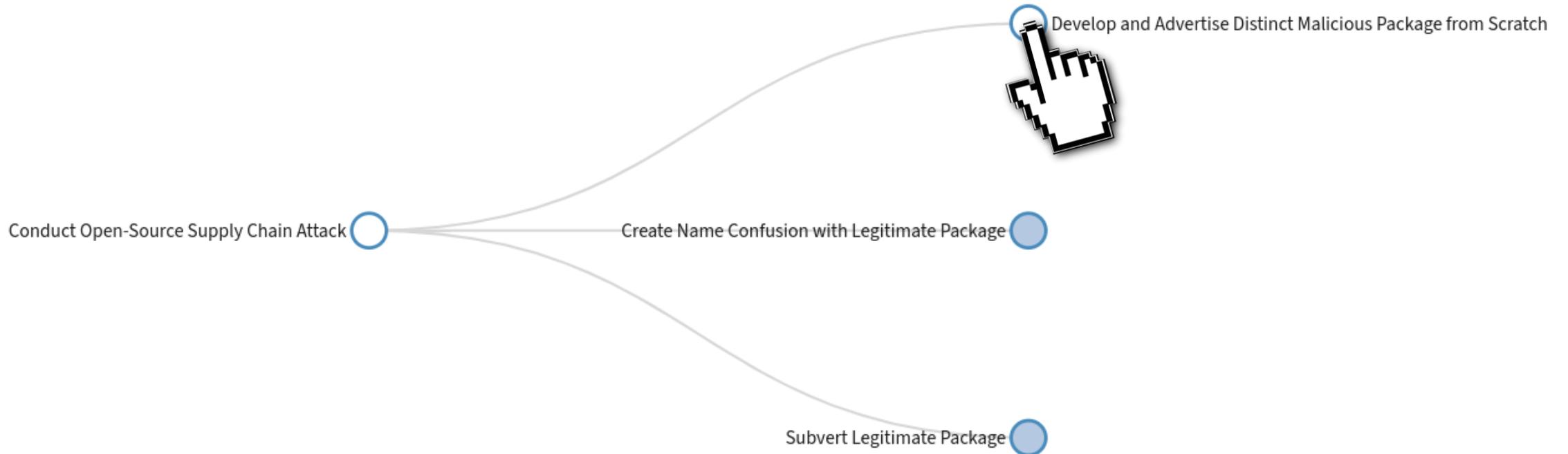
В центре экрана — панель с фильтрами: Всего 13, Критический 6, Средний 4, Низкий 1, Инфо 2, 0. Ниже — поле для поиска по уязвимости или компоненте и переключатель «ВСЕ КОМПОНЕНТЫ». Список компонентов включает: delegates 1.0.0 MIT (4), minimatch 3.0.4 ISC (2), semver 6.3.0 ISC (2), ansi-regex 5.0.1 MIT (1), autoprefixer 8.7.6 (1), **easy-stack 1.0.1 MIT (1)** (выделено синим), lineate 9000.0.0 MIT (1), ms-react-native 0.46.4 MIT (1). Под компонентом easy-stack 1.0.1 MIT отмечено «Supply Chain риск».

В правой части экрана — панель с описанием уязвимости. Заголовок: «Компоненте назначена оценка: 0.5». Сообщение: «Автор библиотеки недоверенный». Ниже — таблица метрик:

Метрика	Оценка
Популярность ⓘ	1.1
Авторский состав ⓘ	0.0
Активность сообщества ⓘ	1.0
Заинтересованность в безопасности ⓘ	Нет
Библиотека создана недавно ⓘ	Нет
Первая версия подозрительно высокая ⓘ	Нет
Единственный проект этого разработчика ⓘ	Нет
% пул-реквестов не выполняют требование рецензии от двух человек ⓘ	0 %

Разбор атак на цепочку поставок (Supply Chain)

Develop and Advertise Distinct Malicious Package from Scratch
(Разработка и реклама отдельного вредоносного пакета с нуля)

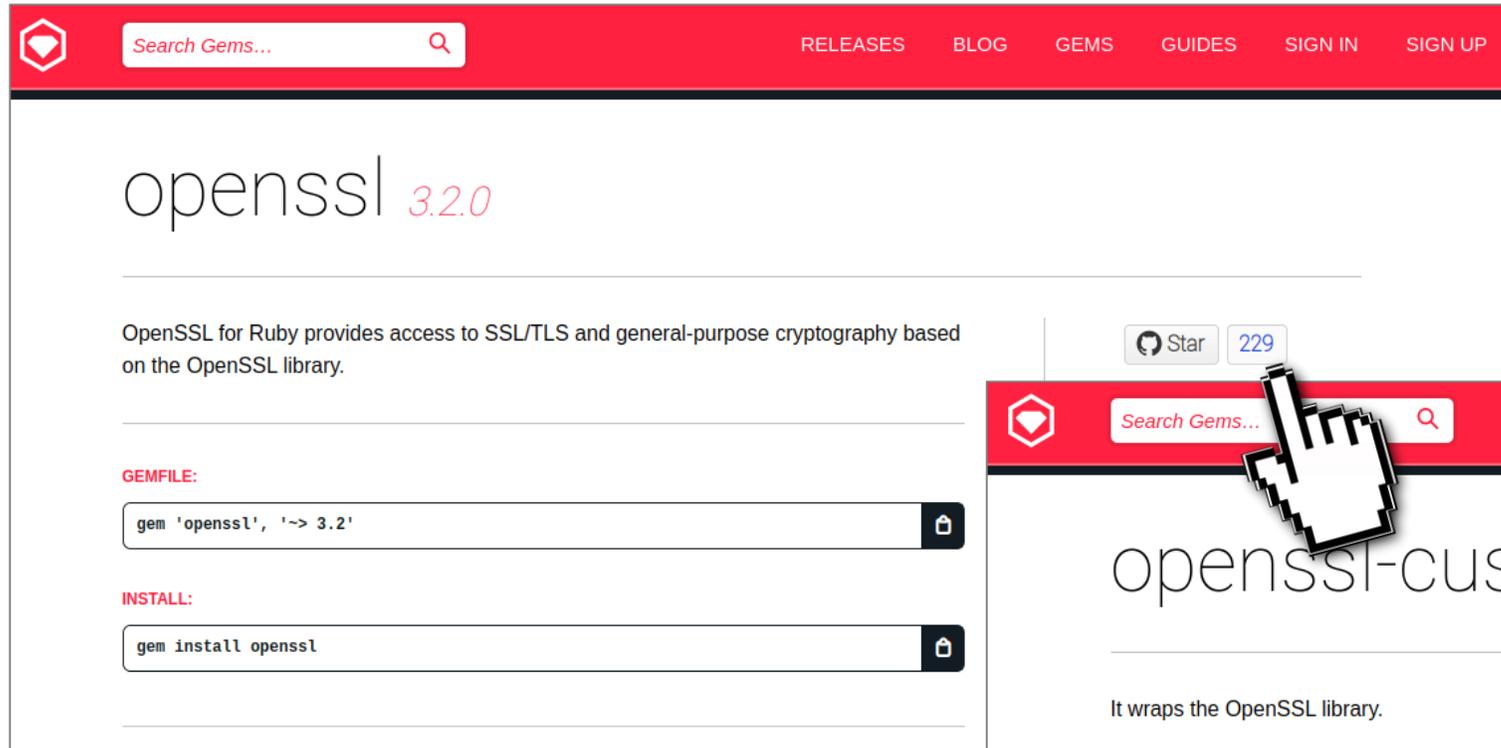


Разбор атак на цепочку поставок (Supply Chain)

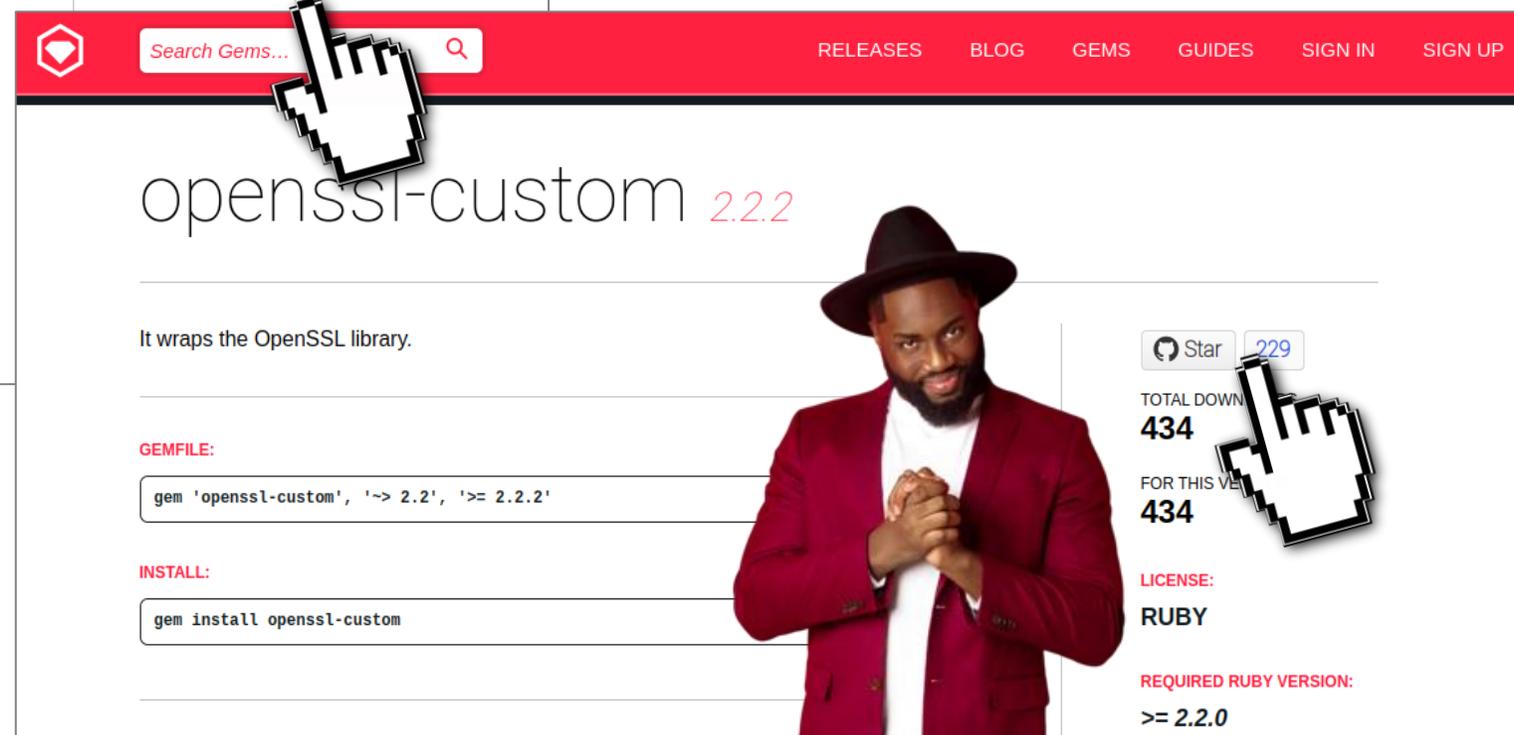
Starjacking - техника актуальна для пакетных менеджеров NPM/Yarn, PyPi и RubyGems. Заключается в краже популярности («звездочек») у чужого известного пакета.



Разбор атак на цепочку поставок (Supply Chain)



The screenshot shows the RubyGems.org page for the 'openssl' gem. The page has a red header with a search bar and navigation links: RELEASES, BLOG, GEMS, GUIDES, SIGN IN, and SIGN UP. The main content area displays the gem name 'openssl' in a large font, with the version '3.2.0' in red. Below the name is a description: 'OpenSSL for Ruby provides access to SSL/TLS and general-purpose cryptography based on the OpenSSL library.' To the right of the description is a 'Star' button with the number '229'. Below the description are two code blocks: one for the GEMFILE and one for the INSTALL command. The GEMFILE block contains the code 'gem 'openssl', '~> 3.2'' and the INSTALL block contains 'gem install openssl'.



The screenshot shows the RubyGems.org page for the 'openssl-custom' gem. The page has a red header with a search bar and navigation links: RELEASES, BLOG, GEMS, GUIDES, SIGN IN, and SIGN UP. The main content area displays the gem name 'openssl-custom' in a large font, with the version '2.2.2' in red. Below the name is a description: 'It wraps the OpenSSL library.' To the right of the description is a 'Star' button with the number '229'. Below the description are three code blocks: one for the GEMFILE, one for the INSTALL command, and one for the LICENSE. The GEMFILE block contains the code 'gem 'openssl-custom', '~> 2.2', '=> 2.2.2'', the INSTALL block contains 'gem install openssl-custom', and the LICENSE block contains 'RUBY'. To the right of the code blocks is a 'TOTAL DOWN' button with the number '434' and a 'FOR THIS VERSION' button with the number '434'. Below these buttons is a 'LICENSE:' section with the text 'RUBY'. At the bottom right is a 'REQUIRED RUBY VERSION:' section with the text '>= 2.2.0'. A large image of a man in a red suit and hat is overlaid on the right side of the page. A hand cursor is pointing at the search bar in the header.

Разбор атак на цепочку поставок (Supply Chain)

Пакет **ms-react-native** присвоил себе репутацию популярного проекта <https://github.com/facebook/react-native> — тип атаки **Starjacking**.

Скриншот интерфейса Solar appScreeener, отображающий результаты сканирования цепочки поставок (SCA) для файла `JS_sbom.json` (ID 9966F0). В верхней панели меню видны пункты: Домашняя страница, Проекты, Группы проектов, Правила и наборы, Администрирование, О продукте. В левом боковом меню: Обзор, Подробные результаты (выделено), Сканирования, Экспорт отчёта, Сравнение сканирований, Настройки.

В центре экрана отображены результаты сканирования:

- Всего: 13
- Критический: 6
- Средний: 4
- Низкий: 1
- Инфо: 2
- 0 (безопасных)

Список компонентов:

- delegates 1.0.0 MIT
- minimatch 3.0.4 ISC
- autoprefixer 8.7.6
- easy-stack 1.0.1 MIT
- lineate 9000.0.0 MIT
- ms-react-native 0.46.4 MIT** (выделено)

Для компонента `ms-react-native 0.46.4 MIT` отмечен **Supply Chain риск**.

В правой части экрана отображены детали уязвимости:

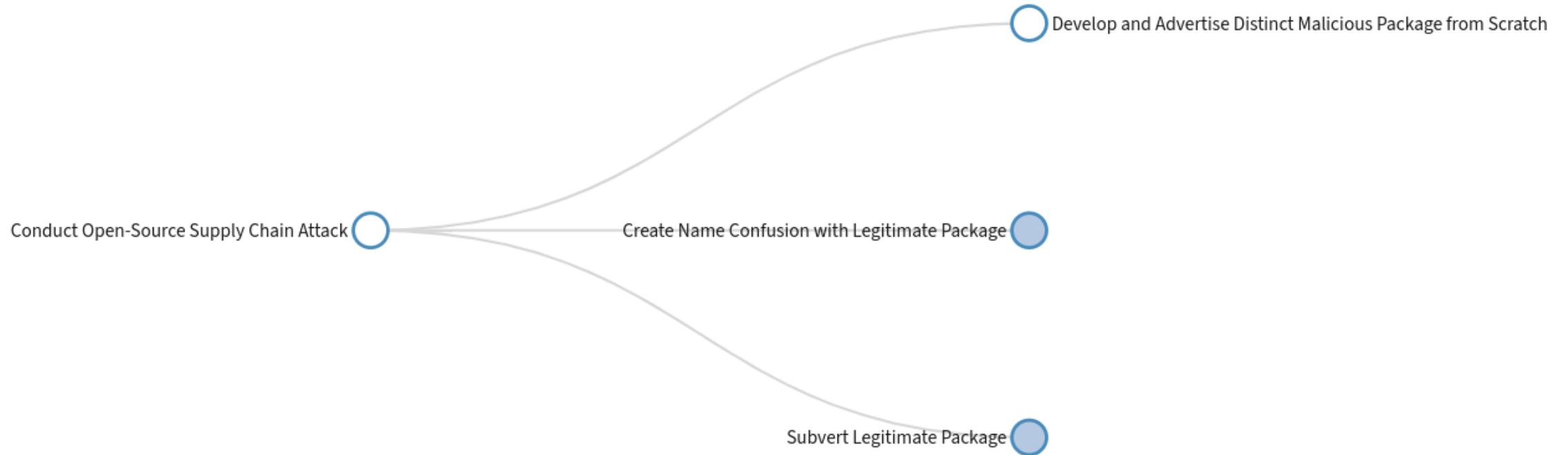
- Описание уязвимости
- Управление уязвимостью
- Таск-менеджер

Компоненте назначена оценка: 0.0.

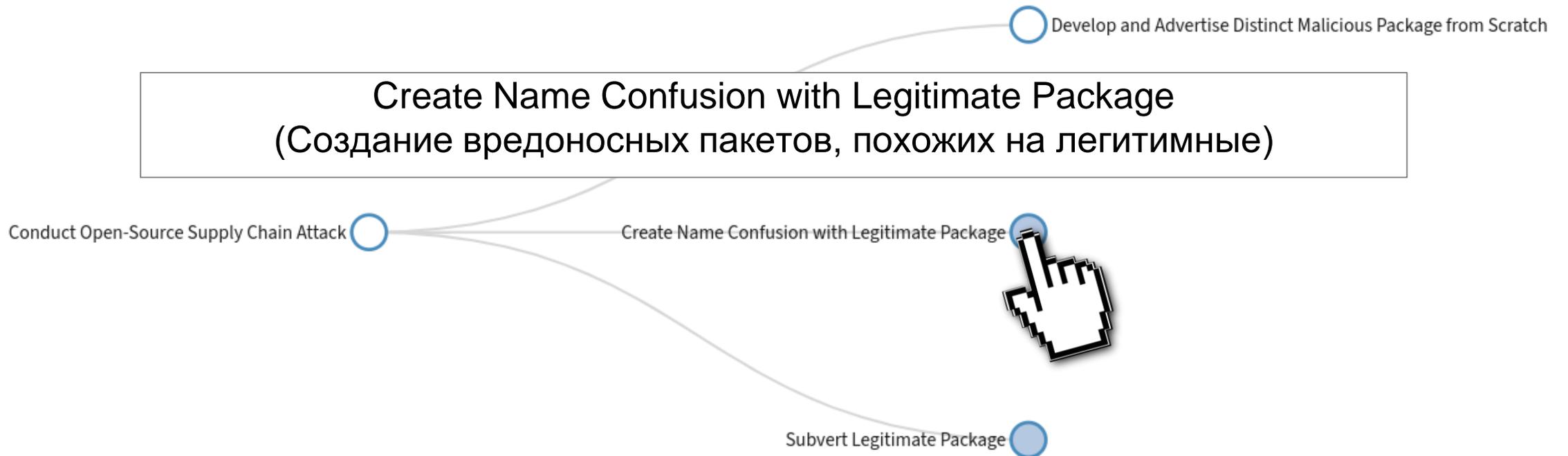
⚠ Данная библиотека использует статистику другой библиотеки.

Метрика	Оценка
Популярность ⓘ	0.0
Авторский состав ⓘ	0.0
Активность сообщества ⓘ	0.0
Заинтересованность в безопасности ⓘ	Нет
Библиотека создана недавно ⓘ	Нет
Первая версия подозрительно высокая ⓘ	Нет
Единственный проект этого разработчика ⓘ	Нет
% пул-реквестов не выполняют требование рецензии от двух человек ⓘ	0 %

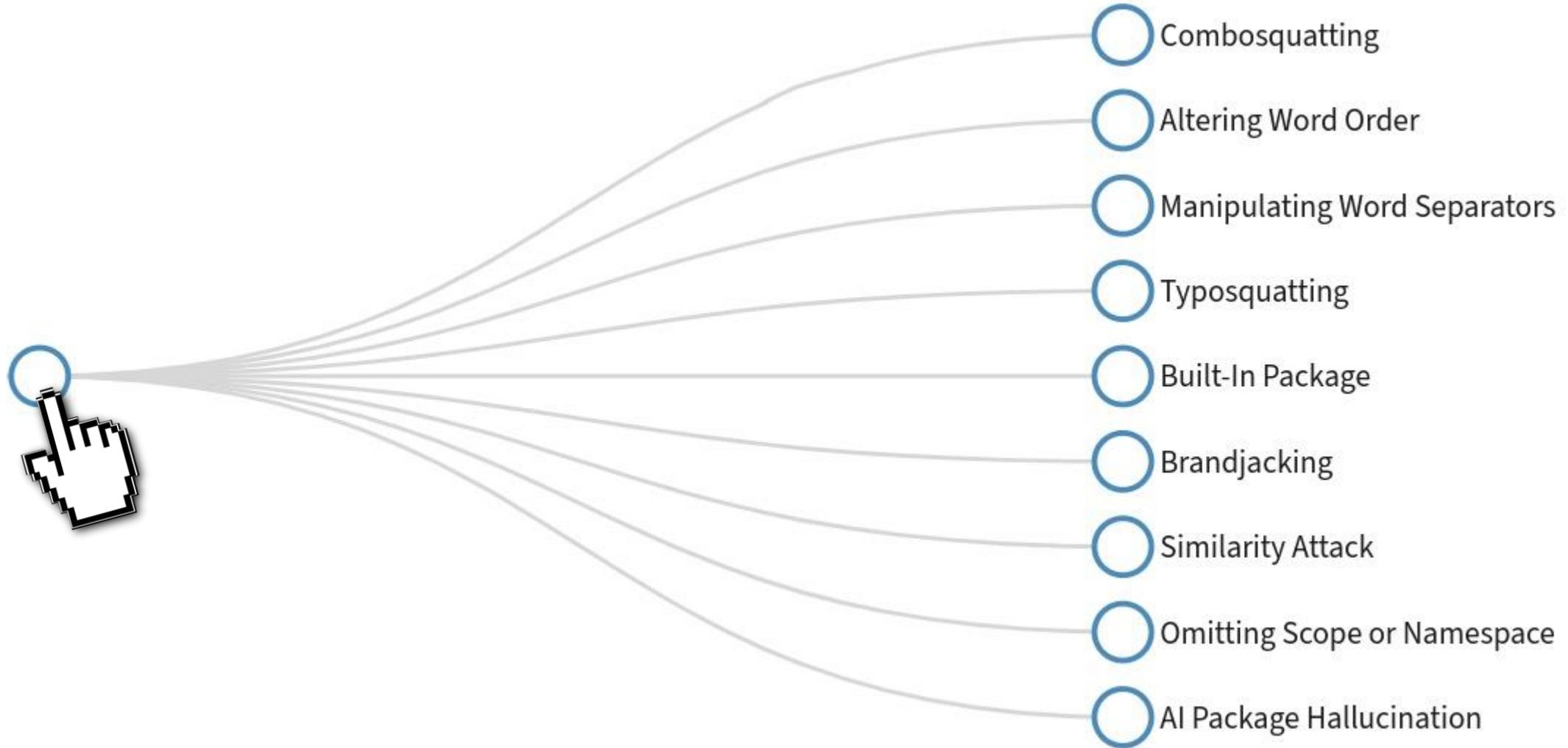
Разбор атак на цепочку поставок (Supply Chain)



Разбор атак на цепочку поставок (Supply Chain)



Разбор атак на цепочку поставок (Supply Chain)



Разбор атак на цепочку поставок (Supply Chain)



colors ^{DT}
1.4.0 • Public • Published 4 years ago

Code ^{Beta} 0 Dependencies 21 436 Dependents 26 Versions

colors.js

Install

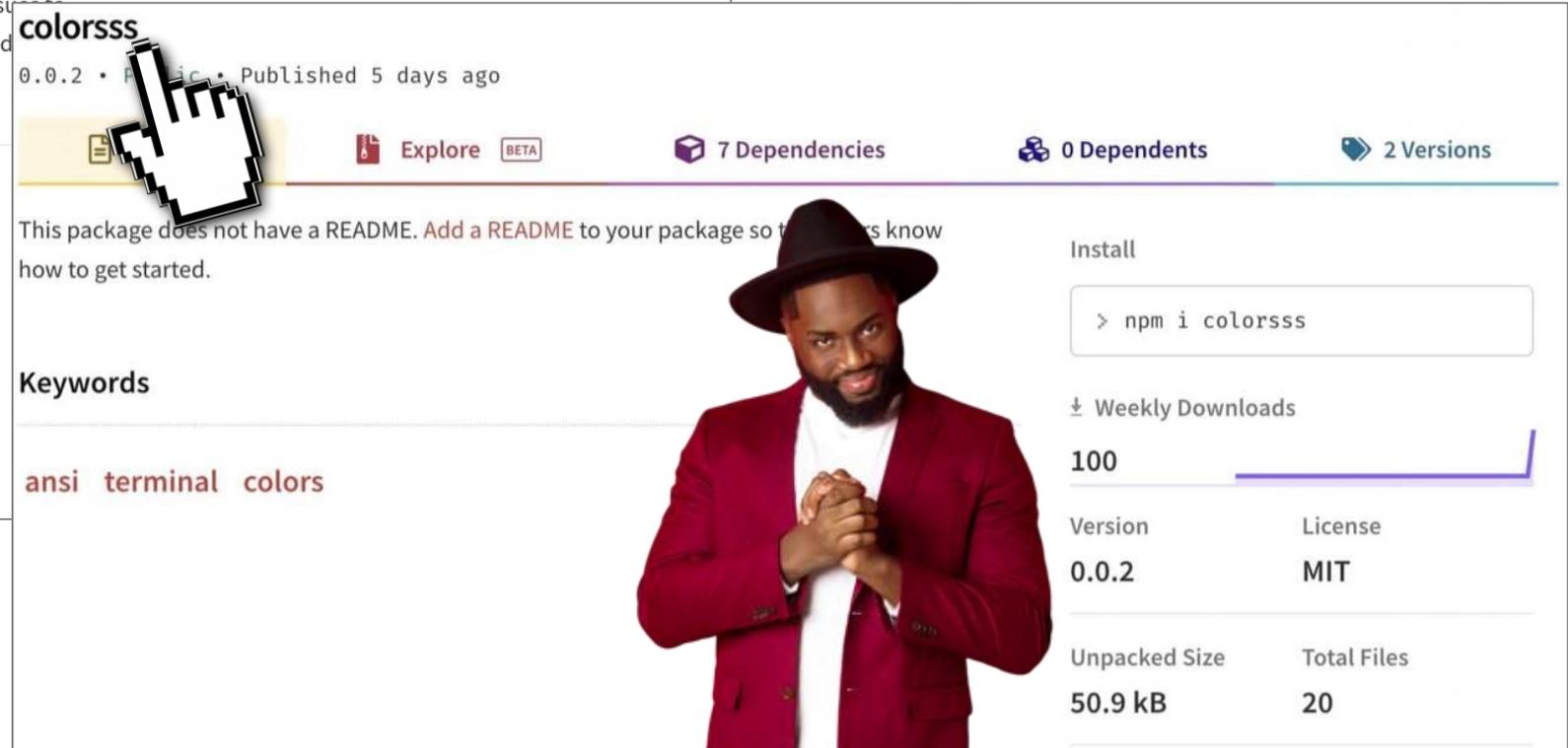
```
> npm i colors
```

build unknown npm v1.4.0 dependencies devDependencies

Please check out the [roadmap](#) for upcoming features and releases. Please open Issues to provide feedback, and check the `develop` branch for the latest bleeding-edge updates.

get color and style in your node.js console

```
→ node examples/normal-usage.js  
First some yellow text  
Underline that text  
Make it bold and red  
Double Rainbows All Day Long  
dR0P THE BASH  
dR0P THE ЯЛ; ηB0ω βΛηθ  
Chains are also cool.  
So are inverse styles!
```



colorsss
0.0.2 • Public • Published 5 days ago

Explore ^{BETA} 7 Dependencies 0 Dependents 2 Versions

This package does not have a README. Add a README to your package so that users know how to get started.

Keywords

ansi terminal colors

Install

```
> npm i colorsss
```

Weekly Downloads

100

Version	License
0.0.2	MIT

Unpacked Size	Total Files
50.9 kB	20



Разбор атак на цепочку поставок (Supply Chain)

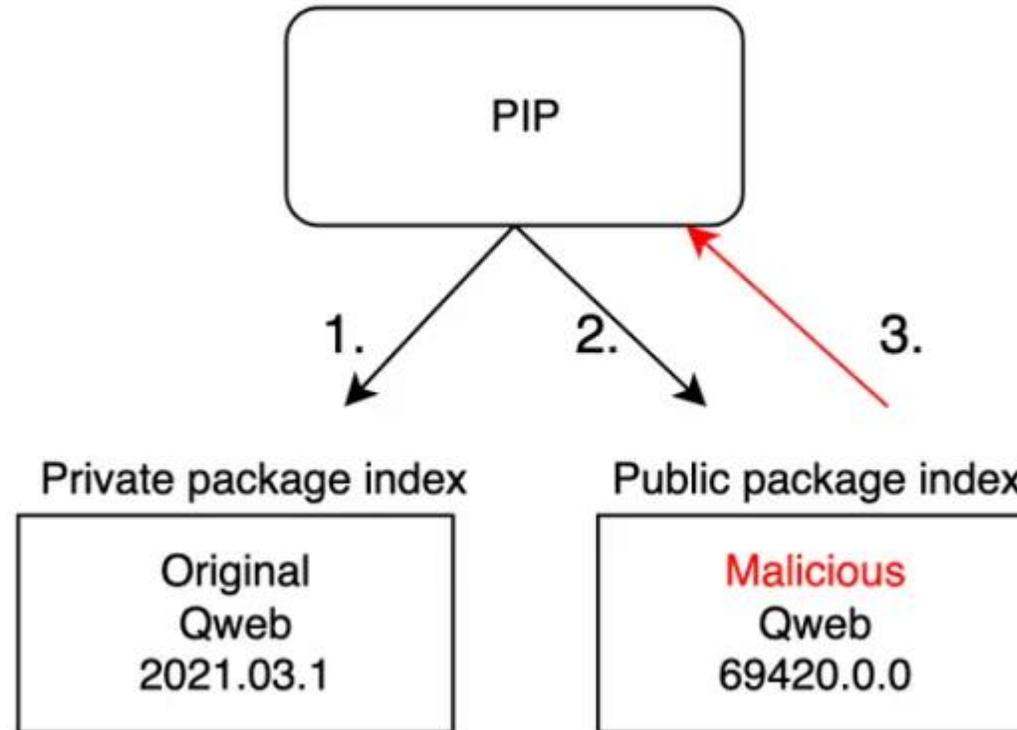
Разработчик опечатался и скачал пакет **autoprefixer** вместо легитимного **autoprefixer**.

The screenshot shows the Solar appScreeener interface. The top navigation bar includes 'Домашняя страница', 'Проекты', 'Группы проектов', 'Правила и наборы', 'Администрирование', and 'О продукте'. The main content area is titled 'Проекты SCA > JS_sbom.json > Подробные результаты'. A summary bar shows: Всего 13, Критический 6, Средний 4, Низкий 1, Инфо 2, and 0 vulnerabilities. A search bar is present with the text 'Поиск по уязвимости или компоненте'. A list of components is shown, with 'autoprefixer 8.7.6' highlighted in blue. Below it, a red square indicates a 'Supply Chain риск'. The right panel shows the 'Описание уязвимости' section with the text 'Компоненте назначена оценка: 0.9.' and a warning: 'Возможно допущена опечатка, имя библиотеки похоже на autoprefixer.' Below this is a table with the following data:

Метрика	Оценка
Популярность ⓘ	0.0
Авторский состав ⓘ	5.0
Активность сообщества ⓘ	3.0
Заинтересованность в безопасности ⓘ	Нет
Библиотека создана недавно ⓘ	Да
Первая версия подозрительно высокая ⓘ	Нет
Единственный проект этого разработчика ⓘ	Нет
% пул-реквестов не выполняют требование рецензии от двух человек ⓘ	0 %

Разбор атак на цепочку поставок (Supply Chain)

Dependency Confusion - запутывание менеджера пакета во время скачивания локальных зависимостей. Для этого злоумышленники могут загружать в общедоступный репозиторий вредоносный пакет с тем же именем, что и внутренний.



Разбор атак на цепочку поставок (Supply Chain)

Dependency Confusion - запутывание менеджера пакета во время скачивания локальных зависимостей. Для этого злоумышленники могут загружать в общедоступный репозиторий вредоносный пакет с тем же именем, что и внутренний.

Neutral Pumpkin Mews Pro Teams Pricing Documentation

npm Search packages Search Sign Up Sign In

@pizza-hut-us-development/client-core

9999.9.9 • Public • Published 3 months ago

Readme Code (Beta) 0 Dependencies 0 Dependents 1 Versions

This package does not have a README. Add a README to your package so that users know how to get started.

Keywords: none

```
> npm i @pizza-hut-us-development/client-core
```

Weekly Downloads: 4

Version	License
9999.9.9	ISC

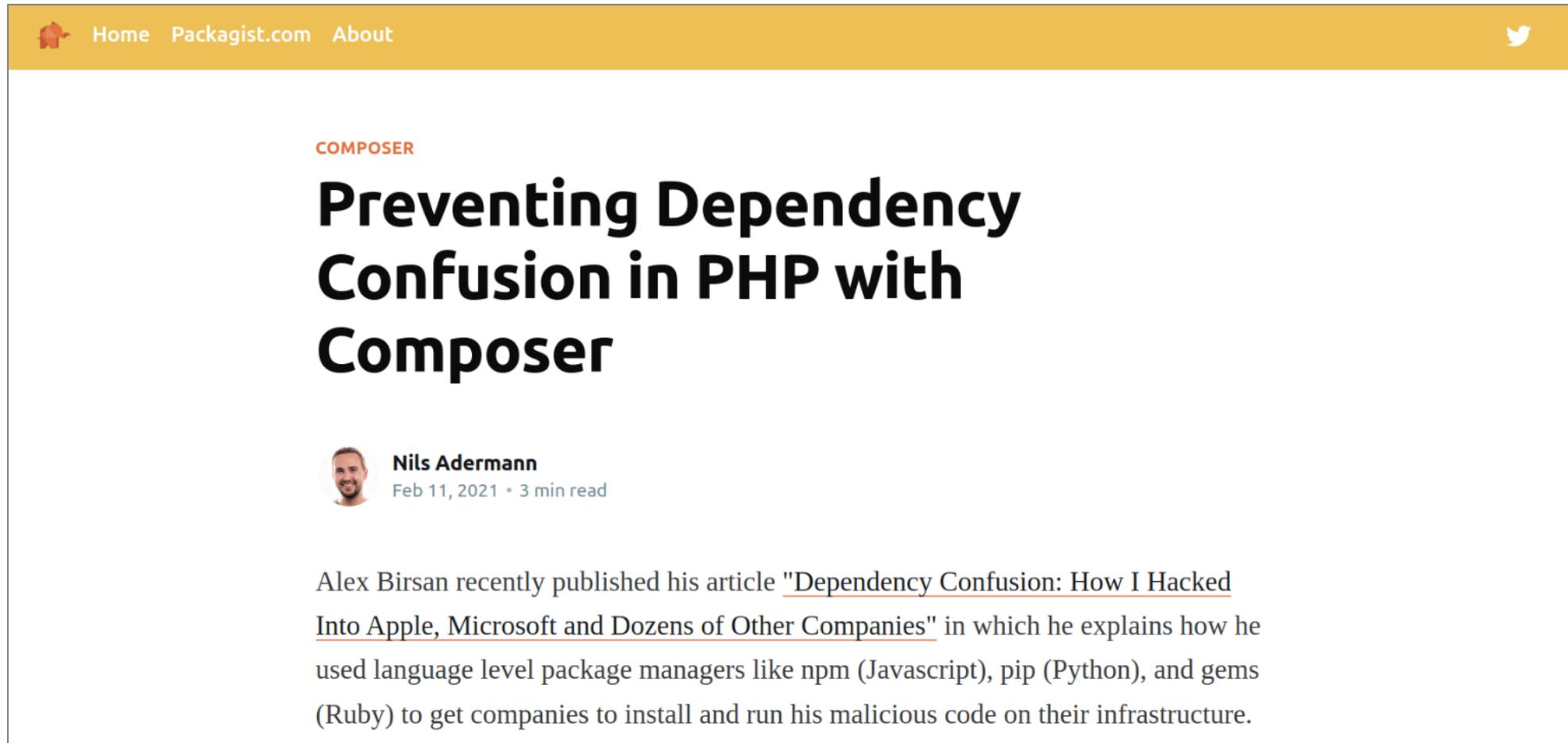
Unpacked	Total Files
391 B	2

Last published: 3 months ago

Collaborators

Разбор атак на цепочку поставок (Supply Chain)

Некоторые пакетные менеджеры самостоятельно задумались о решении этой проблемы на своей стороне. Так, начиная с версии 2.0 **Composer** (пакетный менеджер для PHP), частные репозитории имеют приоритет над общедоступными.



The screenshot shows a web page from Packagist.com. The navigation bar at the top includes a home icon, the text 'Home Packagist.com About', and a Twitter icon. The main content area features a category label 'COMPOSER' in orange, followed by a large, bold title 'Preventing Dependency Confusion in PHP with Composer'. Below the title is a profile picture of Nils Adermann, his name, and the text 'Feb 11, 2021 • 3 min read'. The body of the article begins with a paragraph: 'Alex Birsan recently published his article ["Dependency Confusion: How I Hacked Into Apple, Microsoft and Dozens of Other Companies"](#) in which he explains how he used language level package managers like npm (Javascript), pip (Python), and gems (Ruby) to get companies to install and run his malicious code on their infrastructure.'

Как избежать Dependency Confusion?

Проверить, нет ли уже в общедоступном пакетном менеджере ваших библиотек

Примеры для NPM:

<https://www.npmjs.com/package/YOUR-PACKAGE-NAME>

<https://npmjs.com/org/SCOPE-NAMES>

Заранее зарегистрировать имена своих внутренних пакетов или неймспейсов в общедоступном репозитории. Этот способ **самый эффективный** независимо от того, имеет ли место неправильная настройка сервера или человеческие ошибки

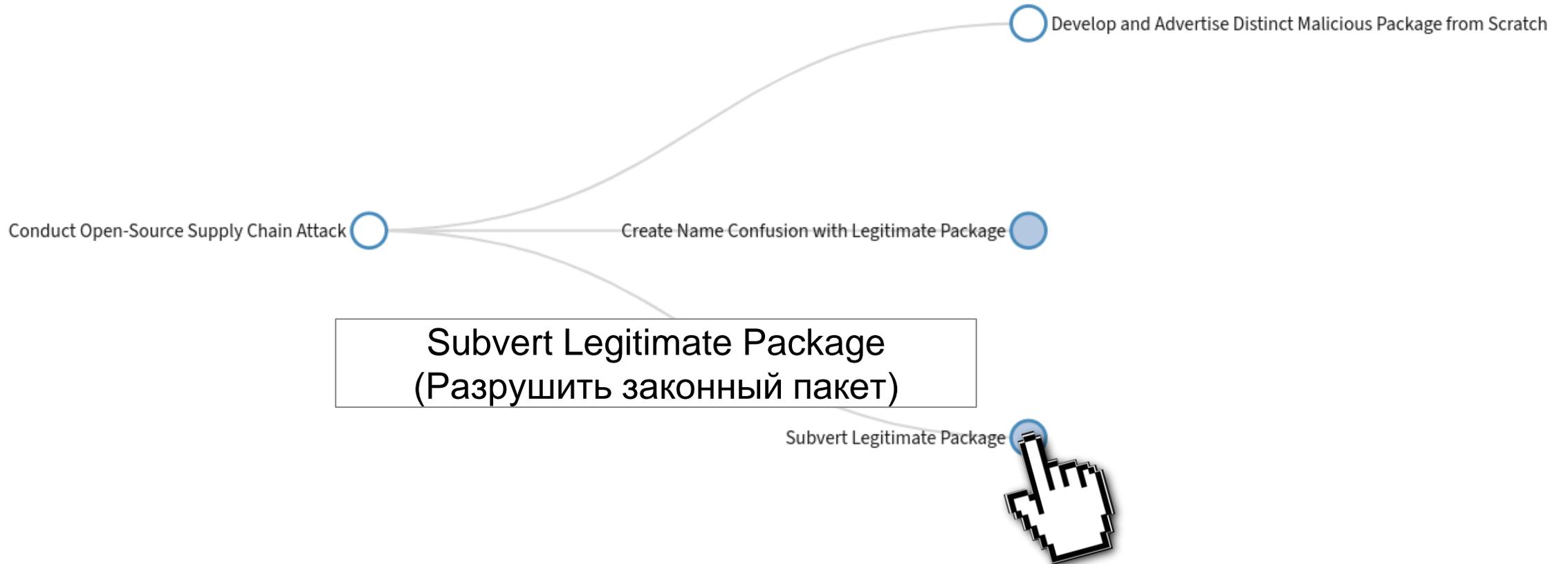
Использовать возможности файлов блокировки для обеспечения безопасного управления зависимостями

Пример: package-lock.json для NPM или Gemfile.lock для Ruby

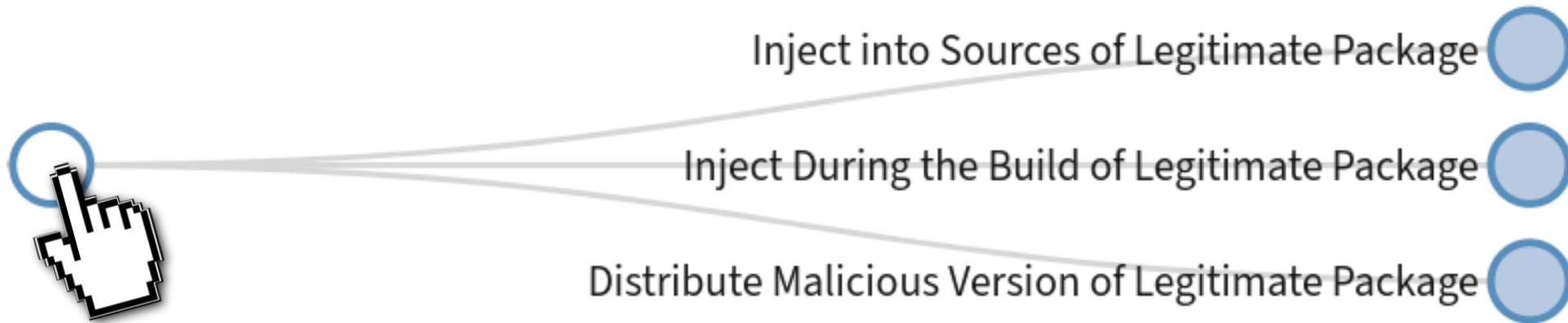
Использовать следующие инструменты:

- Утилита <https://github.com/visma-prodsec/confused> проверяет, заняты ли имена частных пакетов в публичных репозиториях
- Утилита <https://github.com/sonatype-nexus-community/repo-diff> проверяет, нет ли в проксируемых репозиториях Nexus пакетов с такими же именами, что и в частных

Разбор атак на цепочку поставок (Supply Chain)



Разбор атак на цепочку поставок (Supply Chain)



Разбор атак на цепочку поставок (Supply Chain)

[AV-800] **Become Maintainer** - злоумышленник заставляет авторов заброшенных пакетов передать себе права или регистрирует на себя репозиторий под именем другого, который был ранее удален или сменил название.

Разбор атак на цепочку поставок (Supply Chain)

[AV-800] **Become Maintainer** - злоумышленник заставляет авторов заброшенных пакетов передать себе права или регистрирует на себя репозиторий под именем другого, который был ранее удален или сменил название.

event-stream в 2018 г. -
завладели популярной
библиотекой в NPM

 Packt Hub

Malicious code in npm 'event-stream' package targets a bitcoin wallet and causes 8 million download...

Last week Ayrton Sparling, a Computer Science major at CSUF, California disclosed that the popular npm package, event-stream, contains a...

28 нояб. 2018 г.



Оценка риска компрометации через Supply Chain зависимостей

МЕТРИКИ:

- Популярность
- Авторский состав
- Активность сообщества
- Заинтересованность в безопасности
- Библиотека создана недавно
- Первая версия подозрительно высокая
- Единственный проект этого разработчика

Компоненте назначена оценка: 5.0.

Метрика	Оценка
Популярность ⓘ	5.0
Авторский состав ⓘ	5.0
Активность сообщества ⓘ	5.0
Заинтересованность в безопасности ⓘ	Да
Библиотека создана недавно ⓘ	Нет
Первая версия подозрительно высокая ⓘ	Нет
Единственный проект этого разработчика ⓘ	Нет
% пул-реквестов не выполняют требование рецензии от двух человек ⓘ	54 %

Оценка риска компрометации через Supply Chain зависимостей

МЕТРИКИ:

- Популярность
- Авторский состав
- Активность сообщества
- Заинтересованность в безопасности
- Библиотека создана недавно
- Первая версия подозрительно высокая
- Единственный проект этого разработчика

Компоненте назначена оценка: 0.5.

ⓘ Автор библиотеки недоверенный.

Метрика	Оценка
Популярность ⓘ	1.1
Авторский состав ⓘ	0.0
Активность сообщества ⓘ	1.0
Заинтересованность в безопасности ⓘ	Нет
Библиотека создана недавно ⓘ	Нет
Первая версия подозрительно высокая ⓘ	Нет
Единственный проект этого разработчика ⓘ	Нет
% пул-реквестов не выполняют требование рецензии от двух человек ⓘ	11 %

Выводы



Безопасность Supply Chain – это обеспечение безопасности **на всех этапах пути**, по которому ПО попадает в организацию, от момента создания или покупки до использования



Самое уязвимое звено цепочки поставок – **зависимости**



Инвентаризация: **собирать SBOM'ы**.
Вместе с этим проще искать то самое уязвимое звено



Превентивные меры: смотреть, что мы устанавливаем, **до установки**

СПАСИБО ЗА ВНИМАНИЕ!



[Узнать больше о Solar appScreener](#)
[или запросить бесплатную пробную версию](#)

+7 (499) 755-07-70
info@rt-solar.ru

Центральный офис.
125009, Москва,
Никитский переулок, 7с1

