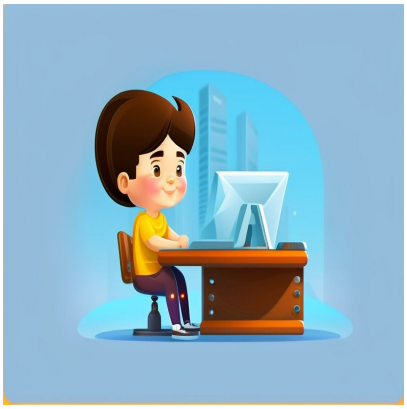


9 Инструментов для защиты NodeJS-приложений от онлайн-угроз

Описание

Node.js, один из ведущих JavaScript-режимов, постепенно завоевывает рынок. Когда что-либо становится популярным в области технологий, оно становится объектом внимания миллионов специалистов, включая экспертов по безопасности, злоумышленников, хакеров и т.д. Ядро node.js безопасно, но при установке сторонних пакетов, способе конфигурирования, установки и развертывания может потребоваться дополнительная безопасность для защиты веб-приложений от хакеров.

Чтобы получить представление, 83% пользователей Snyk обнаружили одну или несколько уязвимостей в своих приложениях. Snyk – одна из популярных платформ для сканирования безопасности node.js. Еще одно последнее исследование показало, что уязвимости были обнаружены в ~14% всей экосистемы npm. В предыдущей статье я рассказывал об обнаружении уязвимостей в приложениях Node.js, и многие из вас спрашивали об их устранении/защите.

Лучшие практики для повышения безопасности Node JS

Ни один фреймворк, включая Node JS, не может быть назван на 100% безопасным. Поэтому, чтобы избежать рисков, необходимо следовать следующим правилам безопасности.

- Регулярно регистрировать и отслеживать действия для обнаружения

уязвимостей
- Не блокируйте цикл обработки событий
- Использовать плоские цепочки обещаний, чтобы избежать ошибок на уровне вложенности
- Создавайте строгие политики аутентификации для своей экосистемы
- Управляйте ошибками для предотвращения несанкционированных атак
- Используйте в своих приложениях анти-CSRF токены
- Предотвращайте утечку данных, отправляя только необходимую информацию
- Правильное управление сессиями с помощью флагов cookie
- Контролируйте размер запроса для предотвращения DoS-атак
- Использовать индивидуальные настройки пакетов и пароль пользователя не по умолчанию
- Внедряйте правила управления доступом для каждого запроса
- Регулярно обновляйте пакеты для защиты от угроз и атак.
- Защита от уязвимостей веб-безопасности с помощью соответствующих заголовков безопасности
- Не используйте опасные функции ради стабильности приложения
- Используйте строгий режим, чтобы избежать ошибок и багов

Теперь мы рассмотрим лучшие инструменты для защиты NodeJS-приложений.

9 Инструментов для защиты NodeJS-приложений от онлайн-угроз

Snyk

Snyk может быть интегрирован в GitHub, Jenkins, Circle CI, Tarvis, Code Ship и Bamboo для поиска и устранения известных уязвимостей. Вы можете понимать зависимости своего приложения и отслеживать в реальном времени предупреждения о нахождении рисков в коде.

На высоком уровне Snyk обеспечивает полную защиту безопасности, включая следующее.

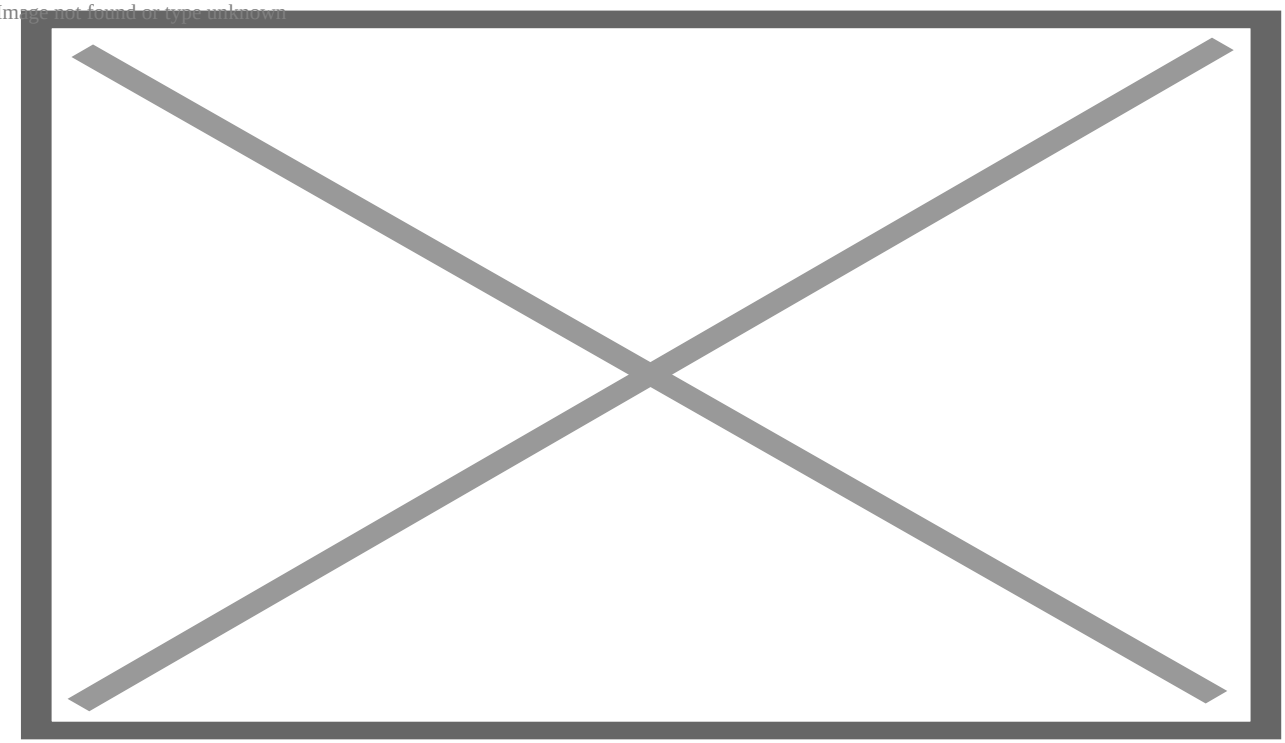
- Поиск уязвимостей в коде
- Мониторинг кода в режиме реального времени
- Исправление уязвимых зависимостей
- Получение уведомлений о появлении новой уязвимости, влияющей на работу приложения.
- Сотрудничество с членами вашей команды

Snyk ведет собственную базу данных уязвимостей и в настоящее время поддерживает Node.js, Ruby, Scala, Python, PHP, .NET, Go и др.

Jscrambler

Jscrambler использует интересный и уникальный подход к обеспечению целостности кода и веб-страниц на стороне клиента.

Image not found or type unknown



Jscrambler делает ваше веб-приложение самозащищенным для борьбы с мошенничеством, позволяет избежать модификации кода во время выполнения и утечки данных, защищает от потери репутации и бизнеса. Еще одна интересная особенность – логика приложения и данные преобразуются таким образом, что их трудно понять и они скрыты на стороне клиента. Это затрудняет угадывание алгоритма, технологий, используемых в приложении. Среди возможностей Jscrambler можно выделить следующие:

- Обнаружение, уведомление и защита в реальном времени
- Защита от инъекций кода, подмены DOM, атак типа “человек в браузере”, ботов, атак нулевого дня
- Предотвращение потери учетных данных, кредитных карт, конфиденциальных данных
- Предотвращение инъекций вредоносного ПО

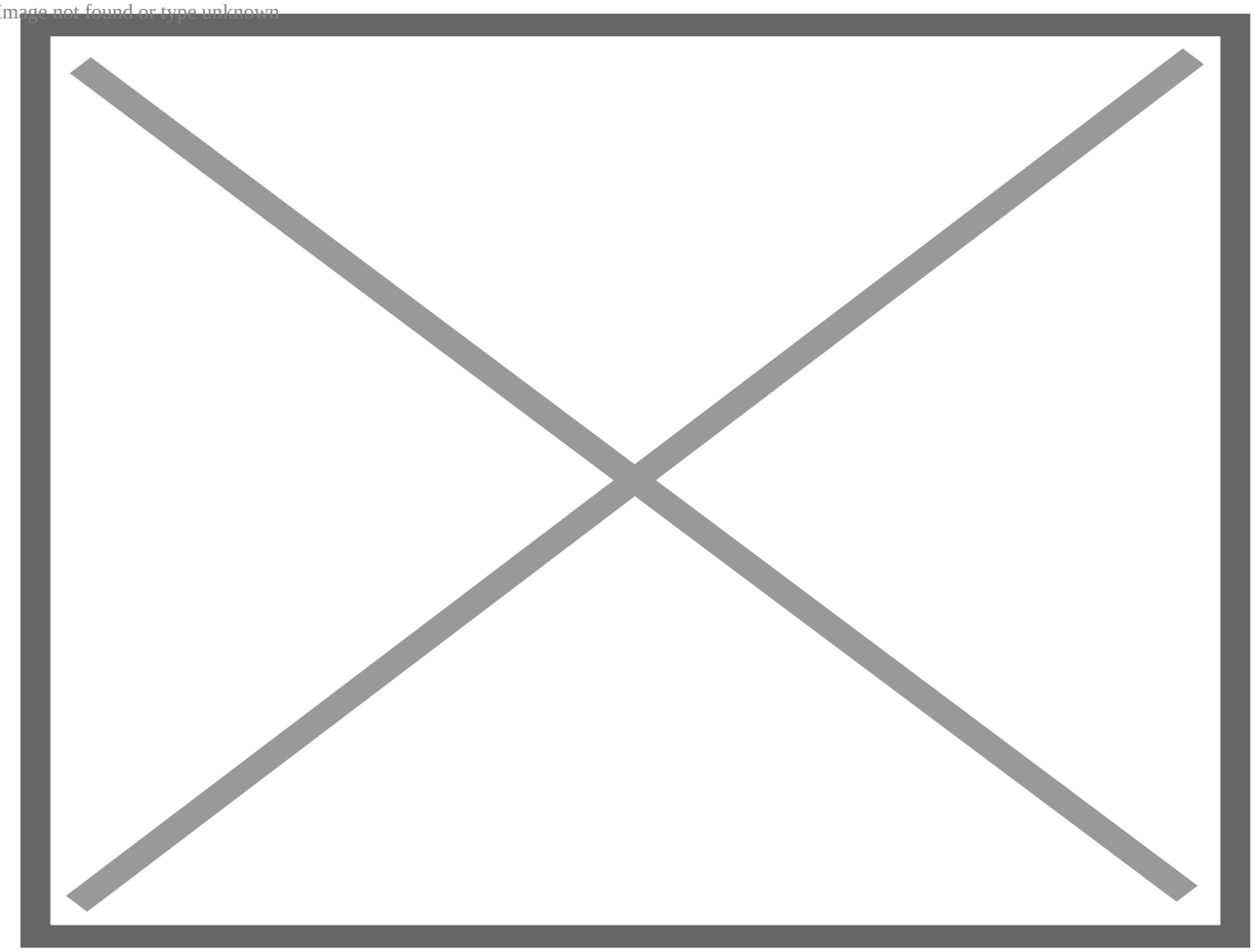
Jscrambler поддерживает большинство JavaScript-фреймворков, таких как Angular, Ionic, Meteor, Vue.js, React, Express, Socket, React, Koa и др. Так что попробуйте, чтобы сделать свое JavaScript-приложение пуленепробиваемым.

Cloudflare WAF

Cloudflare WAF (Web Application Firewall) защищает ваши веб-приложения из облака (на границе сети). Вам не нужно ничего устанавливать в узловое приложение. Существует три типа правил WAF:

- OWASP – для защиты приложения от 10 лучших уязвимостей OWASP.
- Пользовательские правила – вы можете определить правило.
- Cloudflare specials – правила, определяемые Cloudflare на основе приложения.

Image not found or type unknown



Используя Cloudflare, вы не добавляете безопасности своему сайту и пользуетесь преимуществами его быстрой CDN для лучшей доставки контента. Cloudflare WAF доступен в тарифном плане Pro, стоимость которого составляет 20 долл. в месяц. Другим вариантом облачного провайдера безопасности может стать SUCURI и StackPath – комплексное решение для защиты сайта от DDoS, вредоносного ПО, известных уязвимостей и т.д.

Helmet

Сегодня на рынке представлены различные инструменты, и именно поэтому начинающие предприниматели и молодые специалисты путаются в том, какой из них выбрать для конкретной работы. Представляю вашему вниманию Helmet.JS! Helmet основан на модуле Node.JS. Среди его основных задач – повышение безопасности приложений путем настройки HTTP-заголовков и защита от потенциальных онлайн-угроз, таких как Cross-Site Scripting и clickjacking-атаки. Встроенные модули удобны и обеспечивают должный уровень безопасности. Ниже приведены некоторые из модулей, которые мне показались полезными:

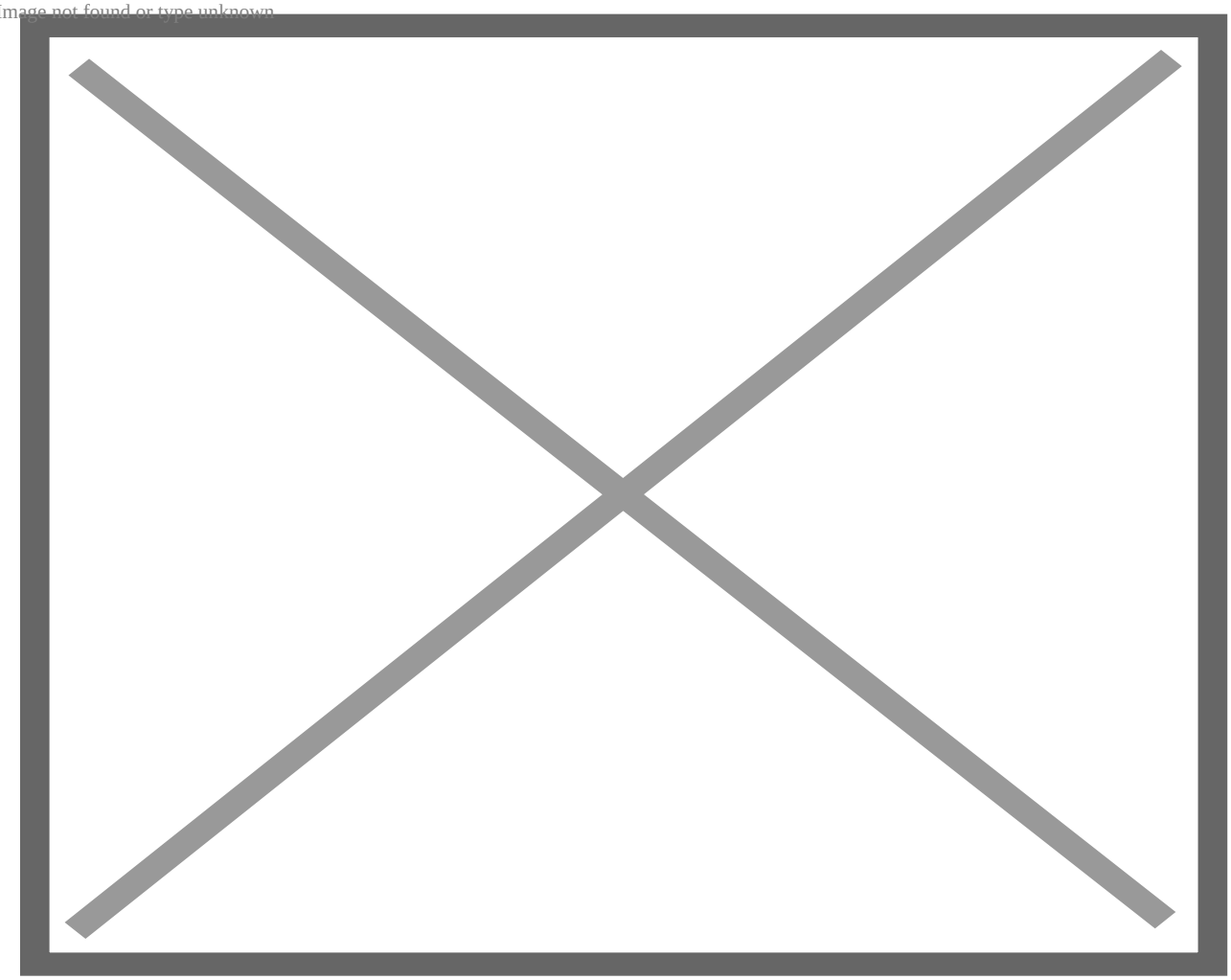
- Content-Security-Policy
- X-Frame-Option
- Public-Key-Pins
- Cache-Control
- Referrer-Policy
- X-XSS-защита

В целом, я считаю, что этот инструмент заслуживает включения в список благодаря тем аспектам безопасности, которые он охватывает.

N|Solid

N|Solid – это готовая платформа для запуска критически важных приложений на Node.js.

Image not found or type unknown



В него встроены функции сканирования уязвимостей в реальном времени и пользовательские политики безопасности для повышения уровня защищенности приложений. Вы можете настроить его на получение оповещений при обнаружении новых уязвимостей в ваших Nodejs-приложениях.

Rate Limit Flexible

Используйте этот миниатюрный пакет для ограничения скорости и запуска функции по событию. Это удобно для защиты от DDoS-атак и атак грубой силы. Некоторые из примеров использования приведены ниже:

- Защита конечных точек входа в систему
- Ограничение скорости краулера/бота
- Стратегия блоков в памяти
- Динамическая блокировка в зависимости от действий пользователя

- Ограничение скорости по IP-адресу
- Блокирование слишком большого числа попыток входа в систему

Интересно, будет ли это замедлять работу приложения? Нет, вы этого даже не заметите. Это быстро, средний запрос занимает 0,7 мс в кластерной среде.

AppTrana Cloud Waap (WAF)

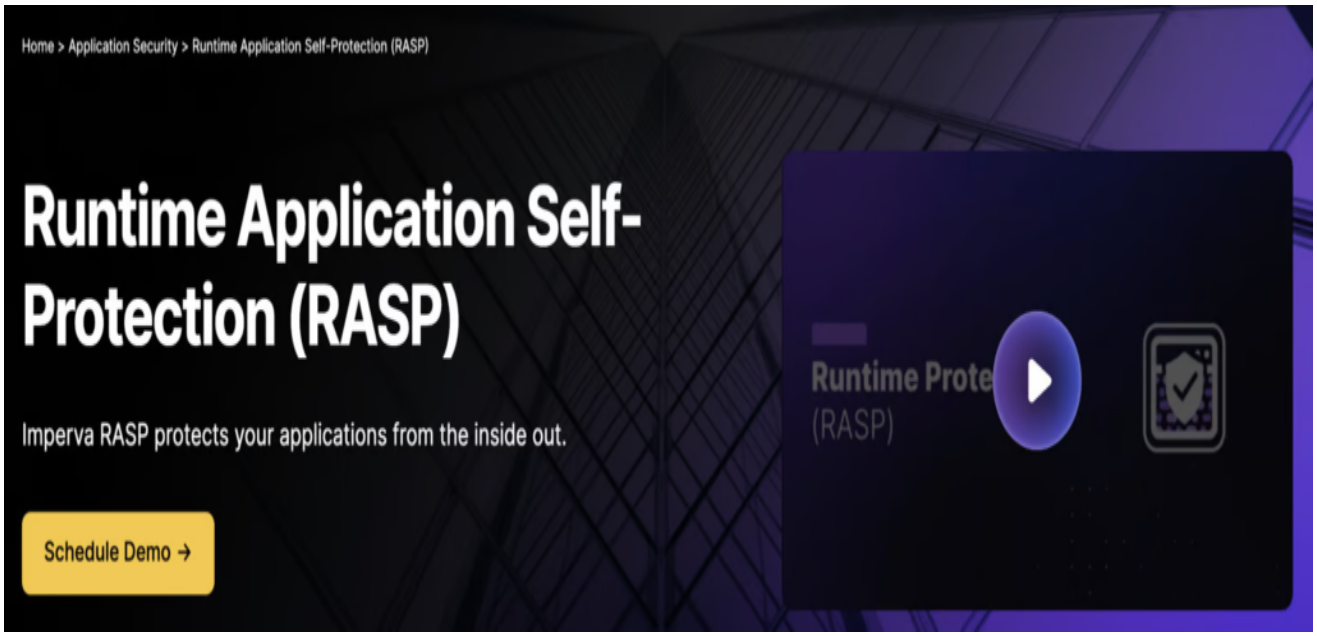
AppTrana рассматривается как полностью управляемое WAF-решение. Оно способно обеспечить комплексную защиту веб-приложений. Оно хорошо известно благодаря своим привлекательным сервисам и возможностям, некоторые из которых приведены ниже:

- **Защита на основе угроз:** Для защиты веб-приложения, как уже говорилось выше, AppTrana использует особый подход, основанный на оценке рисков. Наряду с защитой от ботов она обеспечивает превосходную защиту от API-рисков и DDoS-атак. Кроме того, она помогает обеспечить отличную производительность и безостановочную доступность.
- **Выявление уязвимостей:** Для выявления уязвимостей AppTrana сочетает ручное тестирование на проникновение, в котором участвуют специалисты по безопасности, регулярно проверяющие приложение на предмет выявления потенциальных уязвимостей, и автоматизированные средства сканирования, способные выявлять распространенные угрозы безопасности.

- **Ускорение работы веб-сайтов с помощью безопасной CDN:** Помимо обеспечения безопасности, AppTrana уделяет приоритетное внимание ускорению работы веб-сайтов за счет развертывания сети доставки контента (CDN). Услуги CDN повышают производительность веб-сайтов за счет кэширования контента ближе к конечным пользователям, уменьшения задержек и увеличения времени отклика. CDN в AppTrana построена таким образом, чтобы надежно работать вместе с функциями WAF.

Рассматривая его сервисы и возможности. Я считаю, что этот инструмент заслуживает места в списке. Я рекомендую использовать AppTrana; если вы хотите обезопасить свое приложение и получить желаемые результаты, переходите на AppTrana!

RASP (Runtime Application Self Protection)



Многие организации не успевают за проблемами безопасности и их решениями. Разработаны различные инструменты, помогающие организациям находить уязвимости и лазейки в системе безопасности. В этот список входят инструменты, помогающие организациям и стартапам защитить свои веб-приложения. Среди них есть и “RASP (Runtime Application Self Protection)”! Этот инструмент – отличный вариант для организаций. Он защищает облачные нативные приложения от уязвимостей и обеспечивает безопасность изнутри, гарантируя сохранность приложения. RASP обладает великолепной функцией обнаружения атак, то есть RASP может обнаруживать и защищать от них в режиме реального времени.

Этот инструмент подобен броне, способной защитить от таких атак, как clickjacking, невалидированные редиректы, malformed content types и т.д. Этого не просто достаточно! Он еще и прикрывает вашу спину, предоставляя поддержку по слабым местам ваших веб-приложений. RASP может быть интегрирован с активными приложениями, приложениями сторонних разработчиков, API, облачными приложениями и микросервисами. Честно говоря, мне показалось, что этот инструмент может обеспечить безопасность вашего веб-приложения благодаря двойному эффекту WAF и RASP, что потенциально означает “защиту в глубину”. Его фантастические и столь необходимые функции достаточно привлекательны для стартапов и организаций, чтобы сделать свои веб-приложения безопасными и помочь им легко находить уязвимости.

DOMPurify

Следующий инструмент нельзя назвать быстрым, он просто супербыстрый! Разработчики называют его sanitizer, поскольку это надежный инструмент для защиты Node.js-приложений. DomPurify предотвращает XSS-атаки и другие уязвимости и зарекомендовал себя как новая звезда в сообществе разработчиков. Главной привлекательной чертой этого инструмента является его скорость и простота использования. Он быстро сканирует, обнаруживает и устраняет угрозы безопасности вашего приложения. DOMPurify работает на стороне сервера с Node.js. Поэтому его установка проста и удобна. Чтобы приступить к работе с DOMPurify, необходимо сначала установить “jsdom”. Я бы рекомендовал использовать этот инструмент, если вы хотите повысить уровень безопасности и побороть значительные угрозы безопасности.

Заключение

Надеюсь, что приведенный выше список средств защиты поможет вам обезопасить ваше NodeJS-приложение.

Дата Создания
21.07.2023