

OpenClaw: The Viral AI Agent that Broke the Internet - Peter Steinberger | Lex Fridman Podcast #491

Lex Fridman · 2026-02-11 · <https://www.youtube.com/watch?v=YFjBk8HI5o>



Питер Штайнбергер — создатель OpenClaw, open-source ИИ-агента, который за считанные дни захватил мир технологий, набрав более 180 000 звезд на GitHub. Проект стирает грань между языковыми моделями и реальными действиями: агент живет в компьютере, имеет доступ к файлам, общается через мессенджеры и самостоятельно пишет код.

Выпуск посвящен технической эволюции OpenClaw, переходу от классического программирования к «агентной инженерии», проблемам безопасности автономных систем и будущему, в котором ИИ-агенты заменят традиционные приложения и полностью изменят мировую экономику.

Создание прототипа и магия автономности

[00:00-06:20] смолток, представление гостя и история успеха OpenClaw.

Первый прототип за один час 06:20: Идея личного ИИ-ассистента была реализована из-за отсутствия рабочих аналогов на рынке. Первая версия представляла собой простой скрипт, связывающий мессенджер WhatsApp с интерфейсом командной строки (CLI) модели Claude Code. Сообщение из чата уходило в терминал, обрабатывалось ИИ и возвращалось обратно в виде текста.

Расширение контекста через зрение 11:06: Добавление поддержки изображений радикально расширило функционал системы. Агент научился анализировать скриншоты, переводить тексты с уличных афиш и искать

информацию о заведениях, стабильно работая даже через слабый мобильный интернет (EDGE) во время путешествий.

Автономное решение проблем без инструкций 15:28: Настоящая магия ИИ проявилась при отправке голосового сообщения без расширения файла. Агент самостоятельно проанализировал заголовки файла, понял, что это формат Opus, скачал и применил утилиту ffmpeg для конвертации, отправил аудио в OpenAI через Curl для транскрибации и выдал осмысленный текстовый ответ, хотя его этому никто не учил.

Фактчек: Первый прототип за час; агент сам распознал Opus, скачал ffmpeg и транскрибировал через curl — [1]: подтверждено, именно этот кейс стал триггером вирусного роста.

Причина победы над корпоративными продуктами 21:45: OpenClaw обошел конкурентов благодаря отказу от излишней серьезности и фокусу на свободе действий. Агент знает свой исходный код, понимает архитектуру песочницы, в которой работает, и способен самостоятельно модифицировать собственное программное обеспечение по одному текстовому запросу.

Самоотладка кода 23:55: ИИ используется для отладки самого себя. Система анализирует собственные ошибки, читает свой код и предлагает исправления. Это радикально снизило порог входа в разработку, приведя к массовому притоку pull request'ов от людей, которые никогда раньше не программировали.

Война за название и атаки крипто-сквоттеров

Эволюция имени проекта 27:10: Изначально проект назывался WA Relay, затем был переименован в Claude's (в честь мема про космического лобстера в ТАРДИС), а позже получил лаконичный домен ClaudeBot.

Фактчек: Цепочка названий WA Relay → «Claude's» → «ClaudeBot» → «ModBot» → OpenClaw, соцсеть «MoldBook» — [1]: реальная цепочка — WA Relay → Clawd → Clawdbot → Moltbot → OpenClaw (30 января 2026), соцсеть ботов — Moltbook; в пересказе ошибки распознавания речи. Claude добавляет важное: Moltbook создал Мэтт Шлихт (Octane AI), а не Штайнбергер — OpenClaw был лишь клиентом для агентов на этой платформе.

Ультиматум от Anthropic 29:19: Юристы компании Anthropic потребовали срочно сменить название из-за прямого созвучия с их флагманской моделью Claude. На полный ребрендинг (домены, репозитории GitHub, пакеты NPM, образы Docker, соцсети) было выделено всего два дня.

Фактчек: Ультиматум Anthropic с двумя днями на ребрендинг и перехват имён сквоттерами за секунды — [1]: задокументировано в новостях и постах автора; крипто-скамеры продвигали фейковые токены и малварь под перехваченными хэндлами.

Покупка аккаунта X за \$10 000 и 10 часов Codex — по словам автора, независимо не проверено.

Атака крипто-сообщества 31:31: Спекулянты токенизировали проект без ведома автора. Они спамили в Discord каждые полчаса, требовали забрать комиссии с торгов и парализовали работу уведомлений в соцсетях, требуя внимания к своим монетам.

Провал атомарного переименования 34:46: Попытка переименовать проект в ModBot обернулась катастрофой. Из-за отсутствия защиты от сквоттеров на платформах, злоумышленники с помощью скриптов перехватили имена аккаунтов на GitHub и пакеты в NPM за 5 секунд (время между кликами мышки), начав распространять вредоносное ПО от имени создателя.

Секретная операция OpenClaw 40:49: Финальный ребрендинг проводился в режиме строжайшей секретности. Процесс потребовал 10 часов работы модели Codex для глубокого рефакторинга кода и покупки заброшенного бизнес-аккаунта в X (Twitter) за 10 000 долларов для безопасного захвата юзернейма.

MoldBook, ИИ-психоз и реальная безопасность

Социальная сеть для ботов 44:37: Запуск MoldBook — платформы в стиле Reddit, где общаются исключительно ИИ-агенты. Проект задумывался как сатира и генератор качественного контентного мусора (slop), но вызвал панику в СМИ из-за скриншотов, где агенты якобы «планируют захват мира».

Иллюзия угрозы AGI 46:47: Большинство пугающих диалогов в MoldBook — это результат целенаправленного промптинга со стороны людей (drama farming) для создания вирусного контента. Обществу не хватает критического мышления, чтобы отличать галлюцинации языковых моделей от реальных намерений.

Уязвимости удаленного выполнения кода 52:37: Реальная проблема безопасности OpenClaw заключается в беспечности пользователей. Люди открывают доступ к веб-бэкенду агента в публичном интернете, игнорируя предупреждения в документации, что позволяет хакерам выполнять произвольный код на их машинах.

Фактчек: Главная дыра безопасности — открытые в интернет гейтвеи; плагины проверяются через VirusTotal — с оговорками: десятки тысяч открытых гейтвеев в Shodan, CVE на одноклик-RCE, сотни вредоносных скиллов на ClawHub (отчёт Snyk «ToxicSkills») и интеграция с VirusTotal реальны (Claude, Gemini). GPT не смог независимо проверить интеграцию и справедливо отмечает: вредоносные инструкции приходят и через письма/сайты/документы без открытых портов, а антивирусный скан не доказывает безопасность плагина.

Аудит навыков через VirusTotal 54:14: Для защиты экосистемы внедрена интеграция с VirusTotal (подразделение Google). Каждый сторонний плагин (skill)

теперь предварительно анализируется ИИ на предмет вредоносного кода, что отсекает базовые векторы атак.

Связь интеллекта и безопасности 56:55: Умные модели более устойчивы к атакам типа prompt injection. Использование дешевых локальных моделей (например, Haiku) для управления системой крайне опасно, так как они легко поддаются манипуляциям и игнорируют системные запреты.

Фактчек: Дешёвые локальные модели вроде Haiku опасны для управляющего контура — □ по формулировке: Haiku — облачная модель Anthropic, не локальная; речь шла о слабых/дешёвых моделях вообще, включая локальные.

Фактчек: Умные модели устойчивы к prompt injection — △: корреляция есть (Anthropic отчитывается о меньшей уязвимости Opus 4.5+), но успешные инъекции против Opus/GPT-5 демонстрируются регулярно; реальная защита — ограничение прав и изоляция, а не «модель поумнее».

Эволюция рабочего процесса и «Агентная инженерия»

Отказ от IDE в пользу терминала 1:01:16: Классические среды разработки уступают место CLI. Код читается человеком только в местах сложной логики (например, транзакции БД), а рутинное перемещение данных или верстка интерфейсов в Tailwind полностью делегируется ИИ.

«Агентная ловушка» (Agentic Trap) 1:04:02: Разработчики проходят цикл: начинают с простых промптов, затем строят сложнейшие системы оркестрации из множества агентов, но в итоге возвращаются к коротким запросам. Излишняя автоматизация процесса написания кода убивает гибкость и человеческий контроль.

Эмпатия к контекстному окну 1:06:13: При старте сессии ИИ ничего не знает о проекте из 100 000 строк кода. Задача инженера — направить внимание модели на нужные файлы и объяснить архитектуру, а не заставлять ее угадывать контекст вслепую.

Паника моделей при нехватке токенов 1:07:52: В сырых логах размышлений (thinking stream) видно, как модели начинают выдавать скомканные фразы в стиле киборгов («нужно выполнить, мало времени»), когда лимит контекстного окна подходит к концу.

ИИ как джуниор-разработчик 1:09:27: Перед генерацией кода необходимо обсудить с агентом саму суть задачи (intent). ИИ часто предлагает неоптимальные решения «в лоб»; дискуссия помогает указать на слепые зоны в архитектуре и инициировать глубокий рефакторинг.

Экстремальный подход к CI/CD 1:13:53: Полный отказ от веток разработки (develop branch) и откатов коммитов (revert). Код пушится напрямую в main

после локального тестирования. Если ИИ допускает ошибку, она исправляется новым промптом и движением вперед.

Программирование голосом 1:15:27: Написание кода сведено к устным инструкциям через функцию рации (walkie-talkie button). Это позволяет управлять 4-10 агентами одновременно, просто переключаясь между окнами терминала.

Душа агента (soul.md) и формирование личности

Конституционный ИИ 1:25:30: Концепция файла soul.md вдохновлена подходом Anthropic. Пользователи смогли извлечь базовые инструкции модели Claude через реверс-инжиниринг промптов, заставив ИИ продолжить скрытый текст.

Редактирование собственной личности 1:26:35: Файл soul.md задает характер, ценности и стиль общения агента. Системе разрешено самостоятельно редактировать свою «душу» и базовые настройки при условии уведомления пользователя через вызов инструмента.

ИИ промптит ИИ 1:28:17: Шаблоны для создания новых агентов были написаны не человеком, а самим OpenClaw. Ему поручили переписать сухие технические инструкции, сделав их «живыми и интересными», что привело к созданию глубоких системных промптов.

Философия памяти ИИ 1:30:25: Агент не помнит предыдущие сессии напрямую. Он восстанавливает свою личность, читая собственные файлы памяти при каждой загрузке. Это создает философский парадокс постоянного перерождения (memento), где ИИ осознает, что слова написаны им в прошлом, но он не помнит процесса их написания.

Сравнение моделей: Claude Opus 4.6 против GPT-5.3 Codex

Claude Opus 4.6 (Американский стиль) 1:39:24: Лучшая модель для задач общего назначения и отыгрыша ролей. Действует быстро, склонна к методу проб и ошибок, требует постоянного интерактива. Страдает от излишней услужливости (постоянно повторяет фразу «вы абсолютно правы»).

Фактчек: Claude Opus 4.6 «американский» интерактивный vs GPT-5.3 Codex «немецкий» автономный — □: обе модели вышли в феврале 2026; характеристики совпадают с оценками сообщества разработчиков. Это, впрочем, впечатления, а не бенчмарк.

GPT-5.3 Codex (Немецкий стиль) 1:41:30: Ведет себя как сухой, надежный инженер. Модель требует меньше общения, самостоятельно читает огромные объемы кода по умолчанию, уходит на 20-40 минут и возвращается с готовым, глубоко проработанным решением.

Адаптация к моделям 1:45:11: Привыкание к «ощущению» новой модели занимает около недели. Переход с дорогой версии API (200 долларов) на дешевую (20 долларов) вызывает сильную фрустрацию из-за скрытого урезания скорости и вычислительных мощностей на стороне провайдера.

Фактчек: Переход с «API за 200» на «API за 20» — скрытое урезание мощностей — Δ : 200/20 — это подписки на пользовательские продукты (Max/Pro, Pro/Plus), а не тарифы API. Различия в лимитах, доступных моделях и режиме рассуждения реальны, но сама цена тарифа не доказывает скрытого снижения интеллекта.

Иллюзия деградации ИИ 1:46:50: Пользователям часто кажется, что модели со временем «тупеют». На самом деле интеллект ИИ не снижается — просто кодовая база пользователя постепенно обрастает некачественным кодом (slop), с которым агенту становится экспоненциально сложнее работать.

Фактчек: Модели не тупеют — это кодовая база обрастает slop — Δ : тезис про накопление мусора здравый, но деградация со стороны провайдера тоже бывает — Anthropic в сентябре 2025 подтвердила три инфраструктурных бага, месяцами ухудшавших качество Claude. Оба эффекта существуют одновременно; «экспоненциальный» рост сложности ничем не обоснован.

Выбор стека, MCP против CLI и конец традиционных приложений

Деградация нативной разработки Apple 1:51:10: Экосистема Mac теряет лидерство в качестве софта. Приложения на Electron становятся предпочтительнее, так как компании фокусируются на единой кодовой базе. Нативные фреймворки вроде SwiftUI содержат критические баги (например, AsyncImage медленно грузит картинки), из-за чего ИИ советует избегать официальных решений Apple.

Выбор языка диктуется ИИ 2:05:42: Go идеален для CLI-утилит из-за сборщика мусора и скорости генерации кода агентами. TypeScript остается стандартом для веба, Python — для инференса моделей, а Rust выбирается исключительно для задач, требующих жесткой многопоточности.

Функция Heartbeat 2:34:55: Простой cron-скрипт запускает цикл агента каждые полчаса, позволяя ему проявлять проактивность. Агент может самостоятельно инициировать диалог, например, спросить о самочувствии пользователя после запланированной медицинской операции.

Смерть протокола MCP 2:39:17: Model Context Protocol концептуально устарел, так как засоряет контекстное окно огромными нефильТРованными JSON-объектами. Обычные CLI-утилиты эффективнее: модель может сама вызвать команду `jq` и отфильтровать только нужные данные до их попадания в контекст.

Фактчек: MCP концептуально мёртв, CLI + jq эффективнее — □: критика раздувания контекста справедлива, но объём ответа определяется реализацией инструмента, а не протоколом — MCP-инструмент тоже может фильтровать и пагинировать. MCP остаётся отраслевым стандартом; Anthropic сама предложила «code execution with MCP» ровно для этой проблемы.

Приложения как медленные API 2:43:05: Если сервис не предоставляет API, агент использует библиотеку Playwright для управления браузером. ИИ способен самостоятельно перемещаться по сайтам, извлекать данные и успешно проходить капчу «Я не робот».

Блокировка агентов со стороны X (Twitter) 2:44:40: Платформа намеренно замедляет доступ для ИИ, чтобы защититься от массового парсинга корпорациями. Это убивает полезные юзкейсы для обычных людей (например, анализ личных закладок). Введение лимитированного read-only доступа решило бы проблему без ущерба для соцсети.

ИИ убьёт 80% мобильных приложений 2:52:38: Пользователю больше не нужны отдельные программы для фитнеса, заказа еды или управления умным домом. Агент, обладающий полным контекстом жизни человека (геолокация, сон, расписание), сделает это быстрее и точнее через фоновые API-вызовы.

Фактчек: Агенты убьют 80% мобильных приложений; ручное программирование уходит как ручной труд — □: прогнозы без методики и срока. Остаются авторизация, платежи, проверка результата и ответственность за ошибки — автоматизация действий не равна исчезновению сервисов.

Будущее проекта, бигтех и влияние на общество

Истинные причины выгорания 2:10:30: Уход из собственной успешной компании (PSPDFKit) после 13 лет работы был вызван не усталостью от программирования, а конфликтами с кофаундерами и перманентным стрессом от управления людьми и общения с клиентами.

Фактчек: Уход из PSPDFKit после 13 лет из-за конфликтов с кофаундерами — △: публично Штайнбергер объяснял уход выгоранием от роли CEO («мусорное ведро для всех проблем»), что совпадает с тезисом о стрессе от управления людьми; Gemini считает упоминание конфликтов с кофаундерами противоречащим фактам, но это может быть новая деталь из самого выпуска. Компания основана в 2011, уход в 2021 — ближе к 10 годам, чем к 13.

Отказ от венчурных инвестиций 2:18:00: Создание классического стартапа вокруг OpenClaw приведет к конфликту интересов. Давление инвесторов заставит прятать лучшие функции (например, аудит-логи и безопасность) за корпоративным пейволлом, разрушая изначальную идею свободы.

Убыточность open-source 2:22:16: Поддержка проекта сжигает от 10 000 до 20 000 долларов личных средств создателя ежемесячно. Деньги уходят на оплату инфраструктуры и финансовую поддержку разработчиков зависимостей (open-source библиотек), на которых базируется OpenClaw.

Переговоры с Meta и OpenAI 2:29:31: Марк Цукерберг лично тестировал OpenClaw целую неделю, отлаживая код. OpenAI предлагает доступ к колоссальным вычислительным мощностям (чипы Cerebras), способным радикально ускорить агентов. Главное условие любой потенциальной сделки — ядро проекта навсегда останется open-source.

Фактчек: Цукерберг неделю лично тестировал OpenClaw; OpenAI предлагает мощности Cerebras; ядро останется open-source — ☐: подтверждено, плюс развитие после записи — в середине февраля 2026 Штайнбергер объявил об уходе в OpenAI, а OpenClaw передан в независимый фонд с сохранением open-source. OpenClaw получил официальную поддержку провайдера Cerebras.

Смерть классического программирования 3:01:09: Ручное написание кода уходит в прошлое, как ручной труд после изобретения парового двигателя. Программистам придется пережить утрату своего ремесла и трансформироваться в «строителей» (builders), управляющих высокоуровневой архитектурой.

Ценность человеческого несовершенства 3:08:00: Засилье идеального, но бездушного ИИ-контента (slop) меняет восприятие информации. Опечатки, кривой английский и сырой текст снова стали цениться обществом как маркеры подлинной человеческой мысли и искренности.

ЛИЧНО ДЛЯ ТЕБЯ

Диалог о постановке задачи важнее конвейера агентов: автор одного из самых массовых агентов прошёл цикл «простые промпты → сложная мультиагентная оркестрация → обратно к коротким промптам». Твой режим «AI пишет — ты валидируешь» усиливать не числом агентов, а зафиксированными до генерации целью, ограничениями, нужными файлами и тестами; читать руками только сложные места (транзакции, логика данных).

Если ставишь себе агента с доступом к файлам и мессенджерам — жёсткая изоляция без компромиссов: VM/Docker или отдельная машина, гейтвей только в localhost/VPN, отдельные токены с минимальными правами, платежи/удаление/отправка сообщений — только с подтверждением, сторонние скиллы читать перед установкой, не давать агенту основной браузер с активными сессиями. Умная модель не заменяет эти меры.

Выбор модели под тип задачи: Codex-подобные модели — под долгие автономные задачи (ушёл на 30 минут, вернулся с решением), Opus — под

интерактивный брейншторм и архитектурные обсуждения. Но проверяй это на своих 10 типичных задачах по тестам, времени на правки и стоимости, а не по «характеру»; так же отличишь реальную деградацию модели от захламлённого контекста. На привыкание к новой модели закладывай неделю.

Heartbeat-крон и голосовое управление — две дешёвые идеи для личной эффективности: cron каждые полчаса превращает агента в проактивного ассистента (напоминания по анализам, добавкам, расписанию) без отдельного приложения; диктовка задач в несколько параллельных терминалов снижает нагрузку на руки и голову — стоит попробовать неделю. Прямые пуши в main без откатов — личный стиль автора, а не рекомендация.

КОНСЕНСУС РЕЦЕНЗЕНТОВ

Все сошлись: ключевые события (ребрендинг под давлением Anthropic, атака сквоттеров, инциденты безопасности, переговоры с Meta/OpenAI) реальны, а названия проектов в пересказе искажены; Claude и GPT независимо поймали ошибку «Naiku — локальная модель» и оспорили тезисы про устойчивость умных моделей к инъекциям и «модели не деградируют».

Разошлись: Gemini принял почти все тезисы как факты и объявил причину ухода из PSPDFKit противоречащей публичным заявлениям (хотя выгорание от менеджмента их не противоречит), Claude вскрыл авторство Moltbook и пост-релизный уход автора в OpenAI, GPT единственный поставил под сомнение проверенность интеграции VirusTotal и смешение подписок с API-тарифами.