AIRNITE

ZERO DAY

ВЫПУСК 02/2026 · ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ

Бэкдор в ядре PHP: как злоумышленники изменили исходный код языка веб-разработки

В марте 2021 года в официальный исходный код PHP попали два вредоносных коммита, замаскированных под именами известных разработчиков. Как работал бэкдор и какие выводы должна сделать любая команда разработки.

Опубликовано: 21 ноября 2026 г. Время чтения: 6 мин Автор: Редакция Airnite

// СИТУАЦИЯ В ЦИФРАХ

2
вредоносных коммита
от имени Расмуса Лердорфа и Никиты Попова (php.internals, 28.03.2021)

11
строк вредоносного кода
в ext/zlib/zlib.c, под видом «Fix typo» (GitHub)

0
аутентификаций требовалось
выполнение кода через HTTP-заголовок для любого посетителя (содержимое коммита)

2FA
обязательна для права записи
с тех пор только через организацию PHP на GitHub (php.internals)

PHP лежит в основе множества систем управления контентом, интернет-магазинов и клиентских порталов. 28 марта 2021 года в официальном репозитории исходного кода проекта PHP появились два изменения, которые выглядели безобидно. Одно из них называлось «Fix typo». На деле оно встраивало бэкдор, позволявший выполнить произвольный код на любом сервере с изменённой таким образом версией PHP.

// Что произошло?

Согласно сообщению одного из ключевых разработчиков PHP Никиты Попова в рассылке php.internals, в репозиторий php-src были внесены два вредоносных коммита: один от имени создателя PHP **Расмуса Лердорфа**, другой — от имени самого Попова. По данным ESET, первое изменение заметили при плановом просмотре после коммита и быстро откатили; разработчики из сообщества спросили, что, собственно, делает этот код.

Проект отреагировал радикально. От собственного Git-сервера git.php.net решили отказаться, а прежние зеркала на **GitHub** стали основным источником. С тех пор право записи есть только у членов организации PHP на GitHub, а для этого обязательна **двухфакторная аутентификация**. В тексте коммита упоминалось название «Zerodium» — компании, торгующей уязвимостями; по данным ESET, её глава отверг какую-либо причастность.

// Как устроена атака?

Вредоносный код занимает одиннадцать строк в файле, отвечающем за сжатие zlib. Он считывает HTTP-заголовок с почти незаметным именем **«User-Agentt»** (с двумя t на конце). Если значение начинается со слова «zerodium», PHP выполняет остаток заголовка как программный код. Злоумышленнику было бы достаточно отправить на сайт специально подготовленный запрос — без входа в систему, без пароля.

Как злоумышленники получили возможность записи, прояснило дополнение от 6 апреля 2021 года. Согласно ему, команда больше не считала, что Git-сервер был скомпрометирован. Изменения загрузили по HTTPS с аутентификацией по паролю через базу пользователей master.php.net; возможно, эта база утекла. Команда перешла на современное хеширование паролей, **сбросила все пароли php.net** и заново развернула сервер.

// Кого это касается?

Атака была направлена не на отдельную компанию, а на **цепочку поставок программного обеспечения**: кто изменяет код в источнике, тот добирается до всех, кто затем его использует. То, что бэкдор обнаружили рано, — заслуга внимательных ревьюеров, а не технической защиты. По словам Попова, которые приводит ESET, изменение удалили до того, как оно попало в рабочую эксплуатацию.

Эта схема касается любой компании, которая разрабатывает собственное ПО или использует сторонние библиотеки. Старые самостоятельно обслуживаемые серверы с собственным управлением пользователями, вход по паролю без второго фактора и доверие к известным именам в истории коммитов — именно эти слабые места и были использованы. Имя в коммите не доказывает, кто на самом деле его написал.

// Что делать сейчас?

Меры, принятые в PHP, — хороший образец: централизованно управлять правами записи, сделать двухфакторную аутентификацию обязательной, отключить устаревшие системы, надёжно хранить пароли. Сюда же относится процесс ревью, который серьёзно относится и к небольшим изменениям. Бэкдор был замаскирован под исправление опечатки — именно такие коммиты часто пропускают не глядя.

Кто использует стороннее ПО, должен знать, откуда взят каждый компонент, и получать обновления из определённых источников. Чек-лист ниже сводит главные шаги воедино.

// Чем поможет Airnite?

Airnite разрабатывает **программное обеспечение** с чистой цепочкой поставок: защищённые репозитории, обязательное ревью, подписанные коммиты и проверка зависимостей в сборочном конвейере. В ходе **тестирования на проникновение** мы проверяем, уязвимы ли извне доступы разработчиков, Git-серверы и системы сборки. В рамках **консалтинга** мы показываем вашей команде, как внедрить процессы, благодаря которым подобные манипуляции становятся заметны на ранней стадии.

> Примечание: обзорная статья об инциденте 2021 года, который считается хрестоматийным примером атаки на цепочку поставок ПО. Данные актуальны на 21 ноября 2026 года.

Чек-лист: что сделать прямо сейчас

[+] Выдавать право записи в репозитории кода только через централизованно управляемые учётные записи с двухфакторной аутентификацией; старые серверы и доступы отключить.

[+] Утверждать каждое изменение в основной ветке через pull request по принципу «четырёх глаз» — в том числе «исправления опечаток».

[+] Подписывать коммиты и проверять подписи в сборочном конвейере.

[+] Целенаправленно искать при ревью подозрительные шаблоны: вызовы eval, необычные имена заголовков, код в модулях сжатия или журналирования.

[+] Защищать базы паролей для доступов разработчиков современными алгоритмами хеширования и после инцидентов сбрасывать все пароли.

[+] Инвентаризировать собственные зависимости (SBOM) и получать их только из доверенных, заранее определённых источников.

Чем поможет Airnite

- Разработка и сайты – Сайты и веб-приложения: быстрые, безопасные, многоязычные.
- Пентест и безопасность – Контролируемые атаки с понятным отчётом и планом мер.
- Консалтинг – Взгляд со стороны на ваши планы и проекты.

Насколько защищена ваша цепочка поставок ПО?

Airnite проверяет, у кого есть право записи в ваши репозитории, как утверждаются изменения и какие сторонние пакеты попадают в ваше ПО. Вы получаете чёткий список пробелов и мер — от обязательной двухфакторной аутентификации и подписанных коммитов до проверки зависимостей.

Запросить первичную консультацию · Написать на эл. почту · Через WhatsApp

Контакты

Есть вопросы по статье? Напишите нам в WhatsApp: +49 174 6834842

Читать онлайн: https://airnite.de/ru/zhurnal/bekdor-v-iskhodnom-kode-php/

Источники

Каждый факт в этой статье подтверждён следующими источниками. Первоисточники отмечены.

1. **php.internals · Nikita Popov: Changes to Git commit workflow** · 28 марта 2021 г. · ПЕРВОИСТОЧНИК
   Первоисточник: два вредоносных коммита, подозрение на компрометацию git.php.net, переход на GitHub, обязательная 2FA
   https://news-web.php.net/php.internals/113838
2. **php.internals · Nikita Popov: Update on git.php.net incident** · 6 апреля 2021 г. · ПЕРВОИСТОЧНИК
   Первоисточник: сервер, вероятно, не был скомпрометирован, возможная утечка базы пользователей, сброс паролей
   https://news-web.php.net/php.internals/113981

3. **GitHub · коммит php/php-src c730aa2 «[skip-ci] Fix typo»** · 28 марта 2021 г. · **ПЕРВОИСТОЧНИК**
   Первоисточник: содержимое вредоносного кода в ext/zlib/zlib.c
   https://github.com/php/php-src/commit/c730aa26bd52829a49f2ad284b181b7e82a68d7d
4. **ESET WeLiveSecurity · Backdoor added to PHP source code in Git server breach** · 30 марта 2021 г.
   Исследование безопасности: обнаружение при ревью, объяснение принципа работы, опровержение Zerodium
   https://www.welivesecurity.com/2021/03/30/backdoor-php-source-code-git-server-breach/

Общая редакционная информация, а не юридическая консультация и не индивидуальная проверка ваших систем. Актуально на дату публикации.

Airnite GmbH · Am Waldbach 7 · 77654 Offenburg · airnite.de