

**rus.why\_freebsd**

**Alexandr Kirilov (<https://github.com/alexandrkirilov>)**

## Почему FreeBSD?

### Вступление

У каждой задачи должно быть свое решение которое позволяет добиться максимального результата при минимальных затратах. Данная статья является выжимкой из знаний полученных ранее для поиска сбалансированного решения.

Задача - разработка комплексного решения для построение хранилищ данных с возможностью обеспечения доступа при высоких нагрузках (high-load) к данным большого объема (big-data) при оперативном (от "сейчас" до 2-3 лет) и длительном хранении (от 3 до 15 лет и выше). Решение должно удовлетворять требованиям малых предприятий и обладать возможностью быть масштабируемым для требований больших предприятий.

Материалы статьи основаны на реальных проектах, уже реализованных или как минимум проведены тесты в максимально приближенных к реальным условиям. Язык реализации Erlang.

Теперь более подробно о задаче:

- любое серверное решение обеспечивающее high-load подразумевает большой трафик как income/outcome так и внутри кластера -inner, в некоторых случаях когда требуется сложная логика хранения или обработки данных соотношение может достигать 1(income/outcome):10(inner)
- высокая надежность
- высокая безопасность с возможностью адаптации решения на уровне исходных кодов ОС
- и т.д.

Список свойств данной задачи можно продолжать очень долго. Для понимания выбора достаточно остановиться на этих трех, все остальные менее значимые

### Linux vs FreeBSD

Linux vs FreeBSD - один из самых некорректных способов описания проблемы выбора. Почему? Потому что Linux это только ядро написанное и поддерживаемое Линусом Торвалдсом, а все остальное это сборки на основе ядра - семейство Linux. Все линейки ОС основанные на этом ядре можно проследить на изображении по этой ссылке Linux Distribution Timeline.

FreeBSD - в отличии от Linux, полноценная операционная система. Корректное сравнение должно выглядеть приблизительно так:

- Ubuntu vs FreeBSD
- Fedora or RedHat vs FreeBSD
- FreeBSD vs SlackWare
- и т.д.

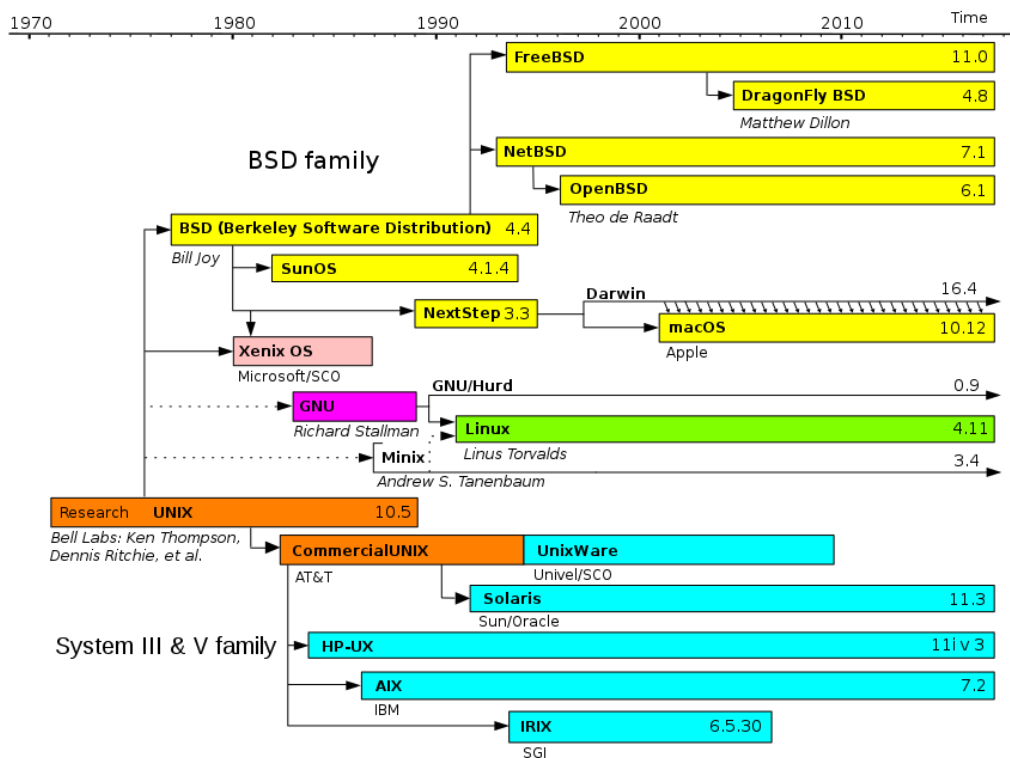
При выборе базового решения для задачи было протестировано несколько операционных систем: FreeBSD, Ubuntu, SlackWare, Fedora. Тесты длились по 2 месяца на каждую систему при одинаковом приложении запущенном на серверах одинаковой конфигурации и одинаковой задаче для высокой нагрузки +10K соединений в секунду.

Если быть честным то до 2K соединений в секунду разницы не было никакой между выше перечисленными системами, что в свою очередь для задач, где не требуется обслуживание такого количества соединений, отдаст предпочтение в сторону Linux-семейства из-за заметной простоты обслуживания и настройки для специалиста среднего или низкого уровня. При выходе на значения свыше +5K соединений на Linux начинались проблемы - в очереди TCP, начинали появляться "длинные ответы" и на +10K становилось очевидным преимущество FreeBSD, разница доходила до 5-7 процентов. При выяснении причин оказалось очень серьезная разница в TCP стеке. Максимально удалось приблизить значения Linux-семейства к значениям на FreeBSD только путем "тюнинга" ядра для SlackWare (обслуживать SlackWare - крайне сложно, очень специфическая сборка Linux), FreeBSD была в состоянии "in-box", производить "тюнинг" не стали. Ubuntu и Fedora на момент проведения теста так и остались внизу списка не смотря на все попытки изменения настроек.

При разработке решений для "тяжелых задач", 5-7 процентов могут быть достаточно весомым аргументом в пользу FreeBSD.

## **История создания и развития**

Общую историю развития UNIX систем можно получить из изображения расположенного ниже.



При развитии решений для хранилищ данных для длительного хранения, немаловажным фактором является скорость разработки операционной системы, она не должна быть очень прогрессивной и новомодной и не должна быть слишком консервативной. В данном случае методология разработки Berkley наиболее оптимальная для решения таких задач.

В процессе эксплуатации FreeBSD и Linux серверов меньше всего было проблем с "наследием" на серверах под управлением FreeBSD. Слишком высокая динамичность может сыграть злую шутку, с тем что вы собираетесь хранить длительное время.

## Безопасность

При обсуждении безопасности во FreeBSD стоит разделить на две категории все вопросы относящиеся к этой теме:

- безопасность самой системы - одна из наиболее редко встречающихся систем в описании уязвимостей, можете проверить это сами [Top 50](#)

Vendors By Total Number Of "Distinct" Vulnerabilities - 27-е место у FreeBSD против 8-го у Linux

- возможности для реализации политик безопасности изначально система ориентировано на уровень больших компаний и государственных органов. При наличии достаточного уровня знаний - можно реализовать ОГРОМНОЕ количество решений не прибегая к закупке дорогостоящих программных решений, "in-box" там есть практически все что нужно

По безопасности, выбор был дан в пользу FreeBSD на основании того что было перечислено выше.

### **Отказоустойчивость**

Среди системных администраторов ходит шутка:

Если ты помнишь как выглядит корпус твоего сервера и где он стоит - он под Microsoft Windows, если ты помнишь наизусть IP адрес сервера и все его настройки - он под Linux, если ты забыл как выглядит сервер и не помнишь его настройки с IP - он под FreeBSD. Почему так? Потому что один раз нормально настроенный сервер может работать годами без остановки.

Данная шутка наиболее ярко иллюстрирует реальное положение дел.

### **САМЫЙ ГЛАВНЫЙ ПУНКТ - LICENSE**

Это самый важный пункт, который в некоторых обстоятельствах может стать ключевым для выбора в пользу FreeBSD.

Между лицензиями GPL и BSD существует огромная разница в виде возможности дальнейшего использования исходных кодов системы. Эта разница звучит так Free Software vs Open Source и где больше свободы на самом деле?

Если быть кратким, то лицензия BSD позволяет делать что угодно и кому угодно с исходными кодами (BSD лежит в основе MacOS, Sony Playstation и большого количества другого ПО обслуживающего критически важные объекты на государственном и военном уровне). Эта политика - своего рода защита разработчиков ПО от данных с которыми ПО может взаимодействовать и позволяет настроить к примеру структуру SSH протокола по своему усмотрению и никому об этом ничего не говорить.

ЛЮБОЕ РЕШЕНИЕ ОСНОВАННОЕ на GNU(Linux) обязывает вас опубликовать все что вы изменили в открытом доступе. Один вопрос: как вы отнесетесь к тому, что вашу систему вскроют не хакеры, а юристы которые придут к вам с ордером обязывающим вас опубликовать все что вы придумали для собственной защиты в открытом доступе?

В ключе разработки решений для хранилищ данных, это может стать краеугольным камнем. Иногда для решения определенных задач нужно вмешиваться в исходный код операционной системы для обеспечения максимальной безопасности и производительности приложения. Если у вас возникает вопрос для чего вмешиваться в исходный код системы? - то это статья не для вас. Вы еще не сталкивались с задачами подобного рода.

### **В итоге ...**

То что было здесь описано рассматривалось как решение для определенной задачи описанной в самом начале. Если вы не видите свой проект с таким уровнем сложности, то нужно очень осторожно отнестись к вопросу о выборе базовой ОС FreeBSD. Помимо огромного числа достоинств этой системы у нее есть один очень серьезный недостаток - простота обслуживания. Для обслуживания и установки сервера под FreeBSD потребуется больше знаний чем для установки любой другой системы.

Следите за обновлениями автора в [\*\*Linkedin\*\*](#).

Следите за AR|BO|RE|US обновлениями в [\*\*Twitter\*\*](#) в [\*\*Linkedin\*\*](#).