Migrating to FreeBSD from a Linux sysadmins perspective
Observation by migrating servers and a laptop for FreeBSD

Albert Dengg
<albert@fsfe.org>

2020-05-18
About Me

▶ Sysadmin
▶ been using mostly Linux for > 20 years
  ▶ mostly Debian, some RedHat based systems, some Ubuntu
▶ relatively new to FreeBSD (5 to 6 years or so)
About Me

- these are obviously my observations at the time of writing/recording this talk
- I might have missed/missused/missunderstood some things...
- conclusions will come at the end :)


Server Migration

Initial Situation

- mixed environment (Windows AD, Windows servers & clients, Linux servers & workstations)
- knowledge of UNIX-like OS by most people is limited
- I want to migrate a few systems to FreeBSD-based systems
Server Migration

Initial Situation

- mixed environment (Windows AD, Windows servers & clients, Linux servers & workstations)
- knowledge of UNIX like OS by most people is limited
- I want to migrate a few systems to FreeBSD based systems
Server Migration

Initial Situation

- mixed environment (Windows AD, Windows servers & clients, Linux servers & workstations)
- knowledge of UNIX like OS by most people is limited
- I want to migrate a few systems to FreeBSD based systems
Server Migration

Tasks

- Making shure that and communicating(!) that the systems can be managed afterwards
- Resolving issues through documentation and asking around
- Keeping the system(s) current
Server Migration

Tasks

- Making sure that and communicating(!) that the systems can be managed afterwards
- Resolving issues through documentation and asking around
- Keeping the system(s) current
Server Migration
Tasks

- Making shure that and communicating(!) that the systems can be managed afterwards
- Resolving issues through documentation and asking around
- Keeping the system(s) current
But we cannot run this when you are not available!
Documentation

- handbook(s)
- manpages
- WIKI
- talking to people
- ...
Documentation

- handbook(s)
- manpages
- WIKI
- talking to people
- ...

...
Documentation

- handbook(s)
- manpages
- WIKI
- talking to people
- ...

...
Documentation

- handbook(s)
- manpages
- WIKI
- talking to people
Documentation

- handbook(s)
- manpages
- WIKI
- talking to people
- ...
Keeping the system(s) current

- base system vs. ports
- freebsd-update
- ports
- pkg
- poudriere
- ...

Keeping the system(s) current

- base system vs. ports
- freebsd-update
- ports
- pkg
- poudriere
- ...

...
Keeping the system(s) current

- base system vs. ports
- freebsd-update
- ports
- pkg
- poudriere
- ...

Keeping the system(s) current

- base system vs. ports
- freebsd-update
- ports
- pkg
- poudriere
- ...
-...
Keeping the system(s) current

- base system vs. ports
- freebsd-update
- ports
- pkg
- poudriere
Keeping the system(s) current

- base system vs. ports
- freebsd-update
- ports
- pkg
- poudriere
- ...
Keeping the system(s) current
Base System vs. Ports

- can be confusing at times if you are used to a more integrated approach
- not really that clean of a separation in practice
  - firmware
  - drivers
Keeping the system(s) current
Base System vs. Ports

- can be confusing at times if you are used to a more integrated approach
- not really that clean of a separation in practice
  - firmware
  - drivers
  - ...
Keeping the system(s) current

Base System vs. Ports

- can be confusing at times if you are used to a more integrated approach
- not really that clean of a separation in practice
  - firmware
  - drivers
  - ...
Keeping the system(s) current

Base System vs. Ports

- can be confusing at times if you are used to a more integrated approach
- not really that clean of a separation in practice
  - firmware
  - drivers
  - ...
Keeping the system(s) current

Base System vs. Ports

- can be confusing at times if you are used to a more integrated approach
- not really that clean of a separation in practice
  - firmware
  - drivers
  - ...
Keeping the system(s) current

freebsd-update

- works decently if it is there
- ... however it’s a bit slow at times
Keeping the system(s) current
freebsd-update

- works decently if it is there
- ... however it’s a bit slow at times
Keeping the system(s) current

Ports

- quite decent collection of software
- compared to some Linux distributions, quite a bit more up2date
- with the quarterly branch there is also an option with less breaking changes
- it can be suboptimal to build everything, especially locally
  - laptops
  - especially graphical applications can take a long time to build
Keeping the system(s) current

Ports

- quite decent collection of software
- compared to some linux distributions, quite a bit more up2date
- with the quarterly branch there is also an option with less breaking changes
- it can be suboptimal to build everything, especially locally
Keeping the system(s) current

Ports

- quite decent collection of software
- compared to some Linux distributions, quite a bit more up2date
- with the quarterly branch there is also an option with less breaking changes
- it can be suboptimal to build everything, especially locally
  - laptops
  - especially graphical applications can take a long time to build
Keeping the system(s) current

Ports

- quite decent collection of software
- compared to some Linux distributions, quite a bit more up2date
- with the quarterly branch there is also an option with less breaking changes
- it can be suboptimal to build everything, especially locally
  - laptops
  - generally graphical applications can take a long time to build
Keeping the system(s) current

Ports

- quite decent collection of software
- compared to some Linux distributions, quite a bit more up2date
- with the quarterly branch there is also an option with less breaking changes
- it can be suboptimal to build everything, especially locally
  - laptops
  - generally graphical applications can take a long time to build
Keeping the system(s) current

Ports

- quite decent collection of software
- compared to some linux distributions, quite a bit more up2date
- with the quaterly branch there is also an option with less breaking changes
- it can be suboptimal to build everything, especailly locally
  - laptops
  - generally graphical applications can take a long time to build
Keeping the system(s) current

pkg

▶ it just works™
▶ unfortunately, not all ports get built for what seem to be mostly licensing/patent reasons (which took me some time to understand)
▶ from what I have read and experienced, mixing building ports and using prebuild packages can lead to problems
▶ at least with the default settings, it can be slow at times
Keeping the system(s) current

pkg

▶ it just works™
▶ unfortunately, not all ports get built for what seem to be mostly licensing/patent reasons (which took me some time to understand)
▶ from what I have read and experienced, mixing building ports and using prebuild packages can lead to problems
▶ at least with the default settings, it can be slow at times
Keeping the system(s) current

pkg

▶ it just works™
▶ unfortunately, not all ports get built for what seem to be mostly licensing/patent reasons (which took me some time to understand)
▶ from what I have read and experienced, mixing building ports and using prebuild packages can lead to problems
▶ at least with the default settings, it can be slow at times
Keeping the system(s) current
pkg

- it just works™
- unfortunately, not all ports get built for what seem to be mostly licensing/patent reasons (which took me some time to understand)
- from what I have read and experienced, mixing building ports and using prebuild packages can lead to problems
- at least with the default settings, it can be slow at times
Keeping the system(s) current
poudriere

- first off: once you understand it, it's quite straightforward
- however, it looks a bit complicated from the onset and the documentation
- downsides from my perspective
  - it seems to be focused on people who want to customize everything
  - which means that at least I did not find anyway to tell it to satisfy build dependencies from prebuild packages
  - which increases build times and resources usages quite a bit (thinking llvm, gcc, ...)

Keeping the system(s) current

poudriere

- first off: once you understand it, it's quite straightforward
- however, it looks a bit complicated from the onset and the documentation
- downsides from my perspective
  - it seems to be focused on people who want to customize everything
  - which means that at least I am one that doesn't use it. The purely build
dependencies from prebuild packages...
  - ...which increases build times and resource usages quite a bit (thinking llvm, gcc, ...)
  - this is important to know when deciding to use it in your projects.
Keeping the system(s) current
poudriere

- first off: once you understand it, it’s quite straightforward
- however, it looks a bit complicated from the onset and the documentation
- downsides from my perspective
  - it seems to be focused on people who want to customize everything
  - ...which means that at least I did not find anyway to tell it to satisfy build dependencies from prebuild packages
  - ...which increases buildtimes and resources usages quite a bit (thinking llvm, gcc, ...)
Keeping the system(s) current

poudriere

➤ first off: once you understand it, it’s quite straightforward
➤ however, it looks a bit complicated from the onset and the documentation
➤ downsides from my perspective
  ➤ it seems to be focused on people who want to customize everything
  ➤ ...which means that at least I did not find anyway to tell it to satisfy build dependencies from prebuild packages
  ➤ ...which increases buildtimes and resources usages quite a bit (thinking llvm, gcc, ...
Keeping the system(s) current

first off: once you understand it, it’s quite straightforward

however, it looks a bit complicated from the onset and the documentation

downsides from my perspective
  - it seems to be focused on people who want to customize everything
  - ...which means that at least I did not find anyway to tell it to satisfy build dependencies from prebuild packages
  - ...which increases buildtimes and resources usages quite a bit (thinking llvm, gcc, ...)

poudriere
Keeping the system(s) current

poudriere

► first off: once you understand it, it’s quite straightforward
► however, it looks a bit complicated from the onset and the documentation
► downsides from my perspective
  ► it seems to be focused on people who want to customize everything
  ► …which means that at least I did not find anyway to tell it to satisfy build dependencies from prebuild packages
  ► …which increases buildtimes and resources usages quite a bit (thinkllvm, gcc, …)
as an experiment (well, partly forced on me), I have been running my main laptop on FreeBSD since BSDCan 2019

this worked quite well on the hardware i had at hand (Thinkpad X220)

over time, some problems emerged:

- sound support in the browser
- some issues with suspend/resume
- some software is unfortunately not available, which lead to running a Linux VM 90% of the time
- is not great for battery runtime
- can be a bit cumbersome at times
as an experiment (well, partly forced on me), I have been running my main laptop on FreeBSD since BSDCan 2019

this worked quite well on the hardware i had at hand (Thinkpad X220)

over time, some problems emerged:

- sound support in the browser
- some issues with suspend/resume
- some software is not usable for reliable which led to running a linux vm 90% of the time
- is not great for battery runtime
- can be a bit cumbersome at times
as an experiment (well, partly forced on me), I have been running my main laptop on FreeBSD since BSDCan 2019

this worked quite well on the hardware i had at hand (Thinkpad X220)

over time, some problems emerged:

- sound support in the browser
- some issues with suspend/resume
- some software is unfortunately not available, which lead to running a linux vm 90% of the time
- is not great for battery runtime
- can be a bit cumbersome at times
Daily Driver/Laptop

- as an experiment (well, partly forced on me), I have been running my main laptop on FreeBSD since BSDCan 2019
- this worked quite well on the hardware i had at hand (Thinkpad X220)
- over time, some problems emerged:
  - sound support in the browser
  - some issues with suspend/resume
  - some software is unfortunately not available, which lead to running a linux vm 90% of the time
- is not great for battery runtime
- can be a bit cumbersome at times
as an experiment (well, partly forced on me), I have been running my main laptop on FreeBSD since BSDCan 2019

this worked quite well on the hardware i had at hand (Thinkpad X220)

over time, some problems emerged:

- sound support in the browser
- some issues with suspend/resume
- some software is unfortunately not available, which led to running a Linux VM 90% of the time
- ... which increased the problems with suspend/resume
- some tools had set up issues running
- some apps would not run at all
as an experiment (well, partly forced on me), I have been running my main laptop on FreeBSD since BSDCan 2019

this worked quite well on the hardware i had at hand (Thinkpad X220)

over time, some problems emerged:

- sound support in the browser
- some issues with suspend/resume
- some software is unfortunately not available, which lead to running a linux vm 90% of the time
  - ... which increased the problems with suspend/resume
- is not great for battery runtime
- can be a bit cumbersome at times
as an experiment (well, partly forced on me), I have been running my main laptop on FreeBSD since BSDCan 2019

this worked quite well on the hardware i had at hand (Thinkpad X220)

over time, some problems emerged:

- sound support in the browser
- some issues with suspend/resume
- some software is unfortunately not available, which lead to running a linux vm 90% of the time
  - ... which increased the problems with suspend/resume
- is not great for battery runtime
- can be a bit cumbersome at times
as an experiment (well, partly forced on me), I have been running my main laptop on FreeBSD since BSDCan 2019

this worked quite well on the hardware i had at hand (Thinkpad X220)

over time, some problems emerged:
  - sound support in the browser
  - some issues with suspend/resume
  - some software is unfortunately not available, which lead to running a linux vm 90% of the time
    - ...which increased the problems with suspend/resume
    - is not great for battery runtime
  - can be a bit cumbersome at times
as an experiment (well, partly forced on me), I have been running my main laptop on FreeBSD since BSDCan 2019

this worked quite well on the hardware i had at hand (Thinkpad X220)

over time, some problems emerged:
  - sound support in the browser
  - some issues with suspend/resume
  - some software is unfortunatly not availible, which lead to running a linux vm 90% of the time
    - ...which increased the problems with suspend/resume
    - is not great for battery runtime
    - can be a bit cumbersome at times
Daily Driver/Laptop

▶ with a new laptop, the problems increased
▶ since the HW is (compatibly) new, i was only able to get WIFI working with -CURRENT
▶ ...which means building from source
▶ with the seperation of base and ports, at time even boot enviroments did not always make it easy to recover from a failed upgrade attempt
▶ since there seem to be no prebuilt packages for the kernel modules from ports for -CURRENT, this meant also building at least some ports from source
▶ ...and having to retrain the boot enviroment to boot from source
▶ ...and having to retrain the boot enviroment to boot from source
▶ ...and having to retrain the boot enviroment to boot from source
▶ ...and having to retrain the boot enviroment to boot from source
▶ ...and having to retrain the boot enviroment to boot from source
▶ this leads back to the problem of mixing different builds
Daily Driver/Laptop

- with a new laptop, the problems increased
- since the HW is (compatively) new, i was only able to get WIFI working with -CURRENT

...which means building from source

- with the seperation of base and ports, at time even boot environments did not always make it easy to recover from a failed upgrade attempt
- since there seem to be no prebuilt packages for the kernel modules from ports for -CURRENT, this meant also building at least some ports from source

- and having to be mindful of building the base as well as the ports from source
- this led back to the problem of mixing different build
with a new laptop, the problems increased
since the HW is (compatibly) new, i was only able to get WIFI working with -CURRENT
...which means building from source
with the separation of base and ports, at time even boot enviroments did not always make it easy to recover from a failed upgrade attempt
since there seem to be no prebuilt packages for the kernel modules from ports for -CURRENT, this meant also building at least some ports from source
...and since both having a poudriere instance for a single laptop to build everything was not that practical (for me)
...and building everything locally even less so on an ultrabook
this leads back to the problem of mixing different builds
with a new laptop, the problems increased
since the HW is (compatibly) new, i was only able to get WIFI working with -CURRENT
...which means building from source
... which means building from source
with the separation of base and ports, at time even boot enviroments did not always make it easy to recover from a failed upgrade attempt
since there seem to be no prebuilt packages for the kernel modules from ports for -CURRENT, this meant also building at least some ports from source
... and since both having a poudriere instance for a single laptop to build everything was not that practical (for me)
... and building everything, even if only current, was too much
... this leads back to the problem of mixing different builds
Daily Driver/Laptop

- with a new laptop, the problems increased
- since the HW is (compatibly) new, i was only able to get WIFI working with -CURRENT
- . . . which means building from source
- with the separation of base and ports, at time even boot environments did not always make it easy to recover from a failed upgrade attempt
- since there seem to be no prebuilt packages for the kernel modules from ports for -CURRENT, this meant also building at least some ports from source
  - . . . and since both having a poudriere instance for a single laptop to build everything was not that practical (for me)
  - . . . and building everything locally even less so on an ultrabook
- this leads back to the problem of mixing different builds
with a new laptop, the problems increased
since the HW is (compatibly) new, i was only able to get WIFI working with
-CURRENT
...which means building from source
with the seperation of base and ports, at time even boot enviroments did not always make it easy to recover from a failed upgrade attempt
since there seem to be no prebuilt packages for the kernel modules from ports for -CURRENT, this meant also building at least some ports from source
...and since both having a poudriere instance for a single laptop to build everything was not that practical (for me)
...and building everything locally even less so on an ultrabook
this leads back to the problem of mixing different builds
with a new laptop, the problems increased
since the HW is (compatibly) new, i was only able to get WIFI working with -CURRENT
...which means building from source
with the separation of base and ports, at time even boot enviroments did not always make it easy to recover from a failed upgrade attempt
since there seem to be no prebuilt packages for the kernel modules from ports for -CURRENT, this meant also building at least some ports from source
...and since both having a poudriere instance for a single laptop to build everything was not that practical (for me)
...and building everything locally even less so on an ultrabook
this leads back to the problem of mixing different builds
Daily Driver/Laptop

- with a new laptop, the problems increased
- since the HW is (compatively) new, i was only able to get WIFI working with -CURRENT
- ...which means building from source
- with the separation of base and ports, at time even boot enviroments did not always make it easy to recover from a failed upgrade attempt
- since there seem to be no prebuilt packages for the kernel modules from ports for -CURRENT, this meant also building at least some ports from source
  - ...and since both having a poudriere instance for a single laptop to build everything was not that practical (for me)
  - ...and building everything locally even less so on an ultrabook
- this leads back to the problem of mixing different builds
THANK YOU