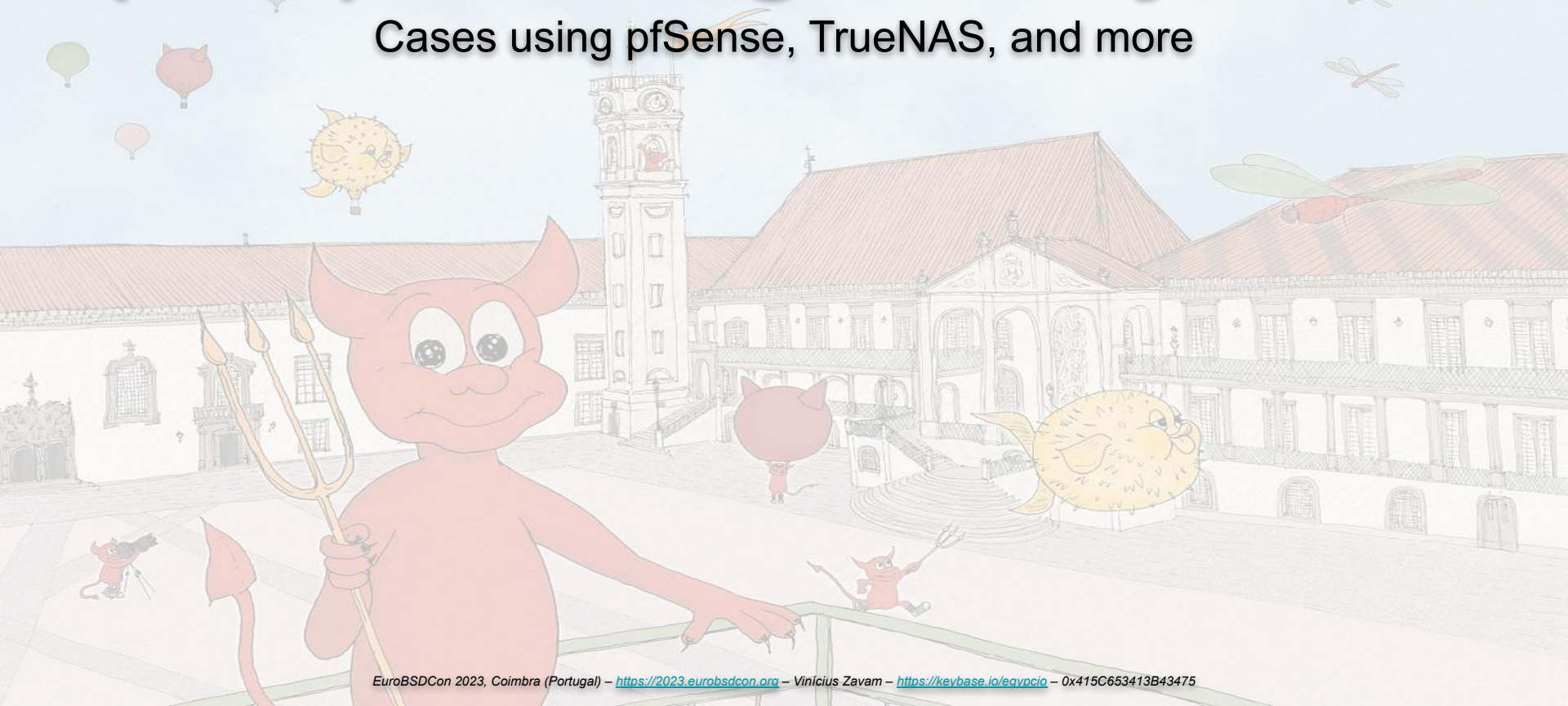


(auto)Installing BSD Systems

Cases using pfSense, TrueNAS, and more



<https://2021.eurobsdcon.org>

tl;dr;

iPXE

(auto)Installers

FreeBSD;

pfSense;

TrueNAS;

ELKE

Demonstration

tl;dr;

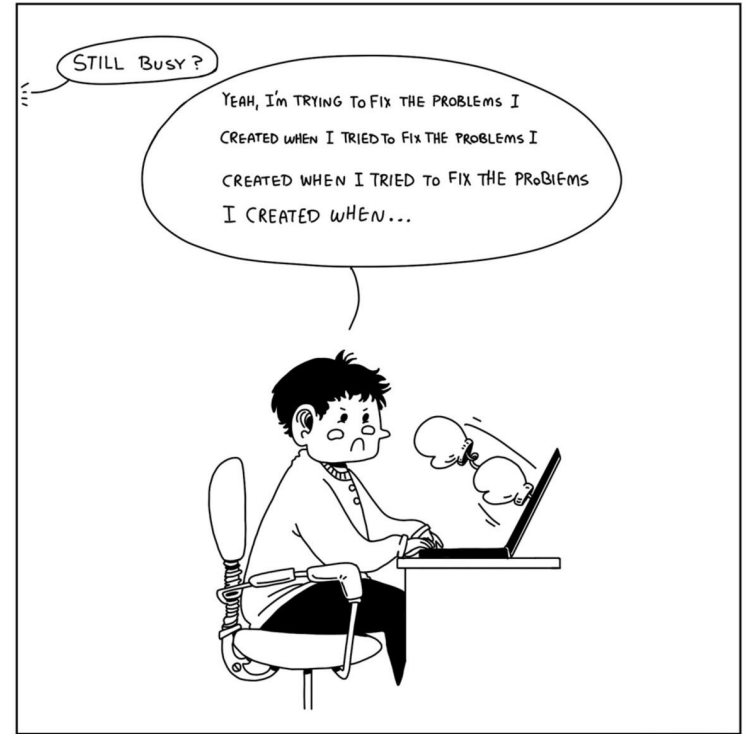
- You can build an alternative solution yourself]=)

– <https://cgkit.freebsd.org/src/tree/release>

`release.sh`

`${TARGET}/make-memstick.sh`

`${TARGET}/mkisoimages.sh`



iPXE

- Full PXE Implementation
 - HTTP(s);
 - IPv6;
 - iSCSI; FCoE; ...
 - VLAN;
- OEM (original equipment manufacturer)
- Expansion ROM
 - Broadcom;
 - Intel;
 - VMware;
- Chain Loading
 - UNDI (universal network driver interface);
 - EFI (extensible firmware interface); UEFI (unified ...);
 - Scripting Support;
- ISO-image or Bootable USB Stick



iPXE

- Cryptography (protocols, ciphers, hashing, ...)
 - TLSv1.2;
 - RSA, RSA/DHE;
 - AES-256-GCM;
 - SHA-512/256;

- Images Trust and Verification

- iPXE supports code signing;
- Verifies the authenticity and integrity of downloaded files;
- <https://ipxe.org/cmd/imgtrust>

- Root Certificates

- In the default configuration, iPXE trusts only the ["iPXE root CA"](#);
- This root CA is used to cross-sign the standard [Mozilla list of public CA certificates](#);
- iPXE will therefore automatically trust the same set of certificates as Firefox;
- You can change the list of trusted root certificates when you build iPXE using the `TRUST` build parameter;
 - FreeBSD's port [net/ipxe](#) allows you to set `IPXE_MAKE_ARGS` for that purpose;
 - ... in a similar way, you can additionally set `IPXE_BUILDCFG` to customize its [buildcfg](#)



iPXE

```
#!ipxe
#
# iPXE; autoexec.ipxe
#
dhcp && route
ntp 10.0.123.11
chain --autofree --replace https://boot.netboot.xyz/ipxe/netboot.xyz.efi
```

iPXE

```
netboot.xyz v2.x - next-server: 192.168.244.254
```

```
Default:
```

```
Boot from local hdd
```

```
Distributions:
```

```
Linux Network Installs (64-bit)
```

```
Live CDs
```

```
Windows
```

```
Tools:
```

```
Utilities (UEFI)
```

```
Architecture: x86_64
```

```
iPXE shell
```

```
Network card info
```

```
About netboot.xyz
```

```
Signature Checks:
```

```
netboot.xyz [ enabled: true ]
```


iPXE

```
#!/ipxe
#
# iPXE; boot.ipxe
#
chain --autofree boot.ipxe.cfg ||
isset ${hostname} && chain --autofree --replace ${boot-dir}/host-${hostname}.ipxe ||
chain --autofree --replace ${boot-dir}/mac-${mac:hexraw}.ipxe ||
chain --autofree --replace ${menu-url} ||
```

iPXE Boot Menu for pxe.localdomain:pfSense

----- BSD Systems (auto)Installers

DragonFlyBSD

FreeBSD

ELKE

pfSense

TrueNAS

NetBSD

OpenBSD

----- Tools and Utilities

Diagnostics and Forensics

Live Operating Systems

pxelinux

Recovery Images

Wipe Hard Drives or Partitions

iPXE Configuration

iPXE Shell

Boot Local Devices

Reboot

Exit

(auto)Installers

- FreeBSD
 - <https://www.freebsd.org/copyright>
 - <https://www.freebsd.org/copyright/freebsd-license>
- pfSense
 - <https://www.pfsense.org/about-pfsense>
 - <https://www.pfsense.org/trademarks.html>
- TrueNAS
 - <https://www.truenas.com/docs/core/gettingstarted/useragreements/coreeula>
 - <https://www.truenas.com/docs/core/gettingstarted/useragreements/enterpriseeula>

(auto)Installers

- [FreeBSD-14.0-CURRENT-amd64-20230323-b5d43972e394-261711-disc1.iso](#)

- diskless(8)

- exports(5);

- nfsd(8);

- pxeboot(8);

- tftpd(8);

- bsdinstall(8)

- bsdconfig(8);

- accounts

- disks partitioning

- networking

- time and date, ...

- /etc/install.cfg

- preamble (env. variables)

- setup



(auto)Installers

```
#  
# FreeBSD; /etc/install.cfg  
#  
export DISTRIBUTIONS="base.txz kernel.txz"  
export PARTITIONS="da0"  
export nonInteractive="YES"  
#!/bin/sh  
sysrc hostname="tortuga.hue.br"  
sysrc ifconfig_vtnet0=DHCP  
sysrc sshd_enable=YES  
sysrc ntpd_enable=YES  
reboot
```



FreeBSD®

(auto)Installers

- [pfSense-CE-2.7.0-DEVELOPMENT-amd64-20230214-0600.iso](https://pfSense.org/en/downloads/iso-images)
 - <https://github.com/pfsense/pfsense>
`recover_configxml.sh`
 - <https://github.com/pfsense/freebsd-src>
`/tmp/buildroom`
`/etc/installerconfig`



(auto)Installers

```
#  
# pfSense; /etc/installerconfig  
#  
export BSDINSTALL_DISTDIR="/usr/freebsd-dist"  
export DISTRIBUTIONS=base.txz  
export FORCE_BOOTMETHOD="UEFI"  
export PARTITIONS=da0  
export PARTMODE="Auto (UFS) UEFI"  
export nonInteractive="YES"
```

(auto)Installers

```
#  
# pfSense; /etc/installerconfig  
#  
export BSDINSTALL_DISTDIR="/usr/freebsd-dist"  
export DEBUG=  
export DISTRIBUTIONS=base.txz  
export FORCE_BOOTMETHOD="UEFI"  
export PARTITIONS=da0  
export PARTMODE="Auto (UFS) UEFI"  
export WORKAROUND_GPTACTIVE=1  
export nonInteractive="YES"  
#!/bin/sh  
touch /cf/conf/___POC  
echo "Secos e Molhados" >> /cf/conf/tropicalia.txt
```


(auto)Installers

- [TrueNAS-13.1-MASTER-202303200713-27864d42a.iso](https://truenas.com/TrueNAS-13.1-MASTER-202303200713-27864d42a.iso)

– <https://github.com/truenas/os>

`/etc/install.sh`

`/etc/install.conf`

– <https://github.com/truenas/core-build>

`/boot/loader.conf`

- mdroot_{name,type}=

- vfs.root.mountfrom=

md(4); mdmfs(8);

makefs(8); mkuzip(8); geom_uzip(4);

mount.conf(5);



(auto)Installers

```
#  
# TrueNAS; /etc/install.conf  
#  
disk=da0  
password=NetBSDover9000malandramente  
whenDone=reboot
```



Elke Maravilha



ELKE (Encrypted and Lovely Kage Environment)

- <https://cgit.freebsd.org/src/tree>

- usr.sbin/bsdconfig

- usr.sbin/bsdinstall

- **INSTALLING**

- *sshd*

- gpart(8); newfs(8); newfs_msdos(8); tar(1); efibootmgr(8);
makefs(8); mkuzip(8); chroot(1); jail(1);
ssh-keygen(1); *sshd_config*(5); sysrc(8); ...

- *elke*

- dd(1); geli(8); gpart(8); zpool(8); zfs(8);
tar(1); sysrc(8); pkg(8); chroot(1); jail(1);
ssh-keygen(1); *sshd_config*(5); ...

ELKE (Encrypted and Lovely Kage Environment)

- **CONFIGURING**

- fstab(5);
- loader.conf(5);
 - kern.geom.eli.tries=0
 - vfs.root.mountfrom="ufs:/dev/gpt/sshd"
- periodic.conf(5);
- pf.conf(5);
- rc.conf(5);
 - root_rw_mount=NO
- security(7);
- tor(1);
 - ClientOnionAuthDir;
 - HiddenServiceDir/authorized_clients/
 - <https://community.torproject.org/onion-services/advanced/client-auth>
- ttys(5);
 - "insecure" console
- zfsprops(7);
 - encryption=aes-256-gcm

ELKE (Encrypted and Lovely Kage Environment)

```
bsdconfig
```

```

Main Menu
-----
If you've already installed FreeBSD, you may use
this menu to customize it somewhat to suit your
particular configuration.  Most importantly, you
can use the Packages utility to load extra '3rd
party' software not provided in the base
distributions.

  █ Exit
  1 Usage
  2 Documentation installation
  3 Packages
  4 Root Password
  5 Disk Management
  6 Login/Group Management
  7 Console
  8 Timezone
  9 Mouse
  A Networking Management
  B Security
  C Startup
  D Ttys

< █ > <Exit bsdconfig> < Help >

```

```
Exit bsdconfig
```

ELKE (Encrypted and Lovely Kage Environment)

FreeBSD Installer

Partition Editor
Create partitions for , F1 for help.
No changes will be made until you select Finish.

da0	24 GB	GPT
da0p1	512 KB	freebsd-boot
da0p2	256 MB	efi
da0p3	4.0 GB	freebsd-ufs
da0p4	20 GB	freebsd-zfs
da0p4.eli	20 GB	
md0	357 MB	preload
md0.uzip	1.1 GB	

[Create] [Delete] [Modify] [Revert] [Auto] [Finish]

ELKE (Encrypted and Lovely Kage Environment)

- **UNLOCKING**

- ssh(1);

- HostKeyAlias; StrictHostKeyChecking; VerifyHostKeyDNS;

- ProxyCommand; ProxyJump; UserKnownHostsFile;

- kldload(8);

- geli(8);

- zpool-import(8); zfs-load-key(8);

- kenv(1); reboot(8);

- vfs.root.mountfrom="zfs:tangamandapio/ROOT/main"*

ELKE (Encrypted and Lovely Kage Environment)

```
mkdir: /tmp/.diskless.b5b34cc345e46cdf8191fbb9a4aad7f5465fc834a2d9caa80c7cf3baff1fb564: Read-only file system
touch: /etc/zfs/exports: Read-only file system
Setting up harvesting: PURE_RDRAND,[CALLOUT],[UMA],[FS_ATIME],SWI,INTERRUPT,NET_HG,[NET_ETHER],NET_TUN,MOUSE,KEYBOARD,ATTACH,CACHED
Feeding entropy: dd: /entropy: Read-only file system
dd: /boot/entropy: Read-only file system
.
add host 127.0.0.1: gateway lo0 fib 0: route already in table
add net default: gateway 192.168.122.1
add host ::1: gateway lo0 fib 0: route already in table
add net fe00::: gateway ::1
add net ff02::: gateway ::1
add net ::ffff:0.0.0.0: gateway ::1
add net ::0.0.0.0: gateway ::1
Mounting late filesystems:.
Performing sanity check on sshd configuration.
Raising kernel security level:
kern.securelevel: -1 -> 0

Fri Aug 11 12:54:26 UTC 2023
2023-08-11T12:54:26.235210+00:00 sshd.localdomain init 1 - - kernel security level changed from 0 to 1

FreeBSD/amd64 (sshd.localdomain) (ttyv0)

login: █
```

ELKE (Encrypted and Lovely Kage Environment)

```
vinicius@zlb: ~  
root@sshd:~ # uname -birmnoUK  
FreeBSD sshd.localdomain 14.0-CURRENT amd64 GENERIC 1400093 1400093 6c1843d69328473aaf42d21fd1e2180a37c57a90  
root@sshd:~ # ps afuxww  
USER PID %CPU %MEM VSZ RSS TT STAT STARTED TIME COMMAND  
root 11 200.0 0.0 0 32 - RNL 13:03 6:07.72 [idle]  
root 0 0.0 0.0 0 656 - DLs 13:03 0:00.04 [kernel]  
root 1 0.0 0.1 11732 1448 - ILs 13:03 0:00.11 /sbin/init  
root 2 0.0 0.0 0 32 - WL 13:03 0:00.06 [clock]  
root 3 0.0 0.0 0 16 - DL 13:03 0:00.01 [md0]  
root 4 0.0 0.0 0 48 - DL 13:03 0:00.00 [crypto]  
root 5 0.0 0.0 0 48 - DL 13:03 0:00.12 [cam]  
root 6 0.0 0.0 0 16 - DL 13:03 0:00.17 [md0,uzip]  
root 7 0.0 0.0 0 16 - DL 13:03 0:00.00 [busdma]  
root 8 0.0 0.0 0 368 - DL 13:03 0:00.03 [zfskern]  
root 9 0.0 0.0 0 16 - DL 13:03 0:00.05 [rand_harvestq]  
root 10 0.0 0.0 0 16 - DL 13:03 0:00.00 [audit]  
root 12 0.0 0.0 0 320 - WL 13:03 0:00.06 [intr]  
root 13 0.0 0.0 0 48 - DL 13:03 0:00.04 [geom]  
root 14 0.0 0.0 0 16 - DL 13:03 0:00.00 [sequencer 00]  
root 15 0.0 0.0 0 80 - DL 13:03 0:00.03 [usb]  
root 16 0.0 0.0 0 48 - DL 13:03 0:00.03 [pagedaemon]  
root 17 0.0 0.0 0 16 - DL 13:03 0:00.00 [vmdaemon]  
root 18 0.0 0.0 0 32 - DL 13:03 0:00.01 [bufdaemon]  
root 19 0.0 0.0 0 16 - DL 13:03 0:00.00 [vnlru]  
root 20 0.0 0.0 0 16 - DL 13:03 0:00.00 [syncer]  
ntpd 452 0.0 0.4 23372 7116 - Ss 13:03 0:00.04 /usr/sbin/ntpd -p /var/db/ntp/ntpd.pid -c /etc/ntp.conf -f /var/db/ntp/ntpd.drift -g  
root 501 0.0 0.6 22564 9268 - Is 13:03 0:00.00 sshd: /usr/sbin/sshd [listener] 0 of 10-100 startups (sshd)  
root 519 0.0 0.6 22800 10136 - Ss 13:03 0:00.13 sshd: root@pts/0,pts/1 (sshd)  
root 518 0.0 0.2 13568 3060 v0 Is+ 13:03 0:00.02 login  
root 522 0.0 0.2 13408 3440 0 Is+ 13:03 0:00.02 -sh (sh)  
root 525 0.0 0.2 13408 3404 1 Ss 13:04 0:00.06 -sh (sh)  
root 543 0.0 0.2 13484 3028 1 R+ 13:06 0:00.00 ps afuxww  
root@sshd:~ # df -Th  
Filesystem Type Size Used Avail Capacity Mounted on  
/dev/ufs/base ufs 1.1G 1.0G 26M 98% /  
devfs devfs 1.0K 0B 1.0K 0% /dev  
tmpfs tmpfs 20M 4.0K 20M 0% /tmp  
tmpfs tmpfs 32M 68K 32M 0% /var  
root@sshd:~ #
```



Demonstration



pfsense®

<https://youtu.be/kMwnRBy6vkQ>



 TrueNAS

<https://youtu.be/J1txpe4YcNU>

"Eu já disse e vou repetir quantas vezes você precisar, como você vai conseguir fazer isso, onde vai procurar e que documentação vai seguir de referência. Pro seu bem, o que eu não vou é colar o comando que você vai simplesmente copiar, executar, agradecer porque funcionou e não terá aprendido nada. Me leia com atenção e você vai descobrir por conta própria exatamente como fazer, agora entregar de mão beijada eu não vou e espero que ninguém faça isso. E acredite em mim é porque gosto de você."

flames > /dev/null

--

saudações,

irado furioso com tudo

Linux User 179402/FreeBSD BSD50853/FUG-BR 154

100% Miko\$hit-free

(auto)Installing BSD Systems

Cases using pfSense, TrueNAS, and more

