PFSense notes:

NAT: State Table
About 10k of RAM for each connection
Bittorrent example with 64MB RAM
If initramfs uses about 6MB and 4MB for web interface, 54MB left * 0.01 = 5400 connections
When State Table get full it drops connections out of RAM and has to re-establish, performance tanks
Is another reason BT clients have option to limit num of connections

32000 states open in a home router is not unusual even when nothing intensive happening

From pfSense Definitive Guide
States - RAM Required
100,000 - ~97 MB
500,000 - ~488 MB
1,000,000 - ~976 MB
3,000,000 - ~2900 MB

Options in pfSense-
Custom traffic shaping - "hierarchical fair service curve"
Fair access + QoS
DNS server/"forwarder" (forwarding meaning caching other DNS servers)
Provisions hostnames.

Installing Squid- cache HTTP objects - Steam example- cached entire download of game
Squid can also be an AdBlocker for phones on Wifi

PfSense can do multiWAN, 10GBe+, Infiniband, etc
Guest networks,
Captive portals
Block domains
Logs

More CPU speed over recommended doesn't do much unless dealing with TONS of traffic
If homebrewing with old boxes, power consumption biggest deal
Before 2.2 pfSense was single-threaded, but since has been multithreaded.

pfSense uses store and forward swithing
RAM speed doesn't matter, DDR2 is fine
Capacity is what matters

Basic can use under 512MB, but over 1GB most installs
Squid caching and Ntop deep packet inspection will use significantly more RAM
Watch the package installs

NICs are important
Intel-based good - small companies cut costs on parts like CPU die/density origins
Also better HW+SW support with Intel-based
PCI cards have limited bandwidth
Each subnet needs another NIC/interface- especially WAN needs separate
If doing something like old laptop can workaround with VLANs but they need management and this gets expensive and welcomes bottlenecks
Wireless drivers not updated well for pfSense
802.11N and AC almost no support out there for Unix/Linux
pfSense can do access control on BSSIDs
VMs for pfSense- if VM/host OS goes down so does network!

Virtualbox - presented is one bridged (for 'WAN') and one intnet connection (for 'LAN')
My ultimate setup was VM bridged using WiFi card and no other NICs (no LAN)
I found this to be true: if pf doesnt see anything but WAN connection, it will put webconfigurator on it, but as soon as LAN shows up, it will restrict to LAN. Doing it this way let me log in on host OS.

Main difference with custom install is the disk partitioning- useful if wanting to make bigger swap
If there are multiple disks, the installer will pick the first in the list to erase
Embeded vs Std kernel - this means console in embedded is only available through serial port
Biggest potential challenge at install time is getting NICs recognized and assigned
If they don't show up on startup to initial pfSense CLI-style menu then use option 1, "assign interfaces"
If cards fail after plugging them in and stuff, it could be a memory buffer problem
In the main startup CLI-style menu, select 8 to get to the shell, then run VI (it doesn't have VIM)
vi /boot/loader.conf.local
[loader.conf.local doesn't get overwritten on update like loader.conf so this is safe]
Add these lines:
kern.ipc.nmbclusters = "131072"
kern.ipc.nmbjumbo9="65536"
Here is the rationale behind the numbers:
Each cluster is about 1.5k, and a reasonable number of these would be 2^17
2^17 is 131072 * 1.5 is 196608 which is about 200MB of RAM
Jumbo frames use a separate buffer
Provision 9k for them 131072*9 is 1.18GB - too big really
2^16 instead is reasonable so entered 65536

The setup wizard asks for a domain name (is also in General Settings).
Don't use a fqdn with TLD unless it is valid. Just call it Ralph or Zeus or something.

NICs Intel generally better but screwed up early 10GB Eth cards so Chelsio recommended
Device polling is for if you have IRQ floods and need it fixed
HW Checksum offload- check disable if card has bad offloading
PowerD under power saving: has to be enabled in BIOS but it is a very good thing.
Settings:
Maximum is almost like not having it on- highest perf values all the time
Minimum runs everything at minimum levels
Adaptive ramps things up when load gets higher, slows when not much to do
Hi adaptive is when performance and interactivity are more important than power consumption.
Hiadaptive will wait before ramping down so it doesn't have to spin up all over again
In a VM the host controls it- not the VM

Crypto- if the hardware is on your system, choose the appropriate to enable it
Same with thermal sensors- if it has them, go for it.

RAM disk is more for running off a sdcard of flash. It manages not being able to do writes as well- it throws logging and stuff in RAM, and dumps in chunks

HD standyby time- just for spinning down of mechanical drives

MSS (under MTU) in WAN interfaces is Maximum Segment Size
Says usually MSS needs to match the MTU of LAN (probably 1500)
In demo, static IP for intnet with MTU 9000 for jumbo and no MSS

"Double NAT'd"
Modem router and cant turn off NAT on firewall - you want "Block private networks" unchecked

When you turn off IP6 on an interface with the above "double LAN" setup you might get a IPv6 complaint about DHCPv6 server set up, and you'd have to turn it off under "services"

The question people usually ask about VPN is "how many connections can my hardware handle?" That is a secondary factor in most deployments, of lesser consideration. The primary consideration in hardware sizing for VPN is throughput required.
The encrypting and decrypting of network traffic with all types of VPNs is very CPU intensive. pfSense offers several cipher options for use with IPsec: DES, 3DES, CAST128, Blowfish 128-256 bit (in 8 bit increments), and AES 128, 192, and 256 bit. The various ciphers perform differently, and the maximum throughput of your firewall is dependent on the cipher used. 3DES is widely used because of its interoperability with nearly every IPsec device, however it is the slowest of all the ciphers supported by pfSense in absence of a hardware crypto accelerator. Hardware crypto accelerators

such as supported cards from Hifn greatly increase maximum VPN throughput, and largely eliminate the performance difference between ciphers. Table 3.4, “IPsec Throughput by Cipher — ALIX” shows the maximum throughput by cipher for PC Engines ALIX hardware (500 MHz Geode) without and with a Soekris vpn1411 Hifn crypto accelerator.

Table 3.4. IPsec Throughput by Cipher - ALIX
Encryption Protocol - Maximum Throughput
DES - 13.7 Mbps
3DES - 8.4 Mbps
Blowfish - 16.5 Mbps
CAST128 - 16.3 Mbps
AES - 19.4 Mbps
AES 256 - 13.5 Mbps

Firewall rules-
Alias allows labeling certain hosts and such
floating rules are for traffic shaping, queue association

Services:
Captive portal, like when a wifi makes you log in or see welcome page first
DHCP server: lots of options, NTP, bootp, etc.
Said both of these will have dedicated overview. I don't see a services video

Places that will take more time to document: traffic shaper wizards, packages, interfaces,
ipsec openvpn and l2tp options

Load balancing service: recall traffic shaping in firewall

status - whats going on with services - dhcp leases a common page to see
diagnostics menu more system-level

Routing- look up types "key"

Diagnostics> Backup and Restore
config.xml has all of pfsense's data aside from rdd data/ logs, package data, but you have the option to back up this as well, or not. (RDD data might be check to exclude from backup, but it is only 4-10MB depending on config and size)
Restore now reinstalls packages for you as well as their data

Config history gives the option to diff the history timestamps
At boot I to install, R for recovery - in recovery you can do this from the CLI

Services>NTP

www.pool.ntp.org
Remember NTP strata levels

This services panel has Serial GPS and PPS settings as well as gen NTP
Access control
Lookup RRD about info- to understand exact definition

Serial GPS to triangulate SAT time can calculate drift - create your own stratum0
GPS sats just broadcast time(!) PPS is pulse per second...

Services> DHCP Server
When you make any changes to the DHCP server, the DHCP server portion of pfSense will restart on saving the new config. Sometimes DHCP doesn't restart up very well- it starts up a new DHCPd when the old one is still shutting down, so binding gets confused, interfaces fail. This may have been fixed, but in the Status>Services panel, shut down DHCP and wait several seconds then start it up

There would be one optionally provided to enable for each interface
Maximum lease time can be set shorter for internet cafes, brief visits, rather than increasing pool
Failover works with CARP settings to step in alt IP

DynDNS subsection if needed
static ARP for static DHCP and ignore other ARP traffic (check on that ignore part)
There is also a section at the bottom for DHCP static mappings which displays those specifically.
At the top of the DHCP section, you can check the box that says "deny unknown clients" and it works with this mechanism. There is an option to additionally just ignore those in the deny list, but it wont work if a failover peer dhcp server is set up. predictably, adding a static mapping will make IP-MAC reservations
The IP addresses you set need to be outside of the pool ranges! this is how you'd specify "excluded addresses a-la Cisco. Here, you interestingly can just reserve the mac address and leave IP empty, and it WILL get an IP from the pool, just not make a MAPPING reservation (arp)

MAC address control offers allow, deny
stats are optional, has a checkbox
NTP options (specify servers including yours on the box), TFTP (bootp source), LDAP to push to clients when queried
Addl bootp options and the network booting portion have very specific info for these settings- **find examples**

DNS Resolver (unbound) vs dnsmasq Forwarder
cache can be used to act as a DNS cache between your devices and internet DNS servers

pfSense uses Unbound - PC asks for url from router listed in it's DNS settings.
That local router asks it's list of DNS servers on net to resolve it
First answer back gets used and cached in router with a tty
Next request sees it in the cache and doesn't do a lookup
Then, the latency is <1 ms instead of 20-400ms

"Unbound is a validating, recursive, and caching DNS resolver. " - https://www.unbound.net
"Dnsmasq is a Domain Name System (DNS) forwarder and Dynamic Host Configuration Protocol (DHCP) server for small computer networks"

https://en.wikipedia.org/wiki/Zone_file

Mentions namebench for benchmarking

https://www.unbound.net/documentation/unbound.conf.html
local-zone: <zone> <type>

Configure a local zone. The type determines the answer to give if there is no match from local-data. The types are deny, refuse, static, transparent, redirect, nodefault, typetransparent, inform, inform_deny, and are explained below. After that the default settings are listed. Use local-data: to enter data into the local zone. Answers for local zones are authoritative DNS answers. By default the zones are class IN.

deny Do not send an answer, drop the query. If there is a match from local data, the query is answered.

refuse
Send an error message reply, with rcode REFUSED. If there is a match from local data, the query is answered.

static
If there is a match from local data, the query is answered. Otherwise, the query is answered with nodata or nxdomain. For a negative answer a SOA is included in the answer if present as local-data for the zone apex domain.

transparent
If there is a match from local data, the query is answered. Otherwise if the query has a different name, the query is resolved normally. If the query is for a name given in localdata but no such type of data is given in localdata, then a noerror nodata answer is returned. If no local-zone is given local-data causes a transparent zone to be created by default.

typetransparent
If there is a match from local data, the query is answered. If the query is for a different name, or for the same name but for a different type, the query is resolved normally. So, similar to transparent but types that are not listed in local data are resolved normally, so if an A record is in the local data that does not cause a nodata reply for AAAA queries.

redirect
The query is answered from the local data for the zone name. There may be no local data beneath the zone name. This answers queries for the zone, and all subdomains of the zone with the local data for the zone. It can be used to redirect a domain to return a different address record to the end user, with local-zone: "example.com." redirect and local-data: "example.com. A 127.0.0.1" queries for www.example.com and www.foo.example.com are redirected, so that users with web browsers cannot access sites with suffix example.com.

inform
The query is answered normally. The client IP address (@portnumber) is printed to the logfile. The log message is: timestamp, unbound-pid, info: zonename inform IP@port queryname type class. This option can be used for normal resolution, but machines looking up infected names are logged, eg. to run antivirus on them.

inform_deny
The query is dropped, like 'deny', and logged, like 'inform'. Ie. find infected machines without answering the queries.

nodefault
Used to turn off default contents for AS112 zones. The other types also turn off default contents for the zone. The 'nodefault' option has no other effect than turning off default contents for the given zone. Use nodefault if you use exactly that zone, if you want to use a subzone, use transparent.

Interface: Adding types

VLAN Configuration
Parent interface - Only VLAN capable interfaces will be shown.
VLAN Tag - default is 1 - 802.1Q VLAN tag (between 1 and 4094).
VLAN Priority - 802.1Q VLAN Priority (between 0 and 7).

Description

QinQ Configuration - ***BROKEN? Cant specify ranges***
Parent interface - Only QinQ capable interfaces will be shown.
First level tag - first level VLAN- On top of this are stacked the member VLANs defined **below.**
Option(s) (checkbox adds interface to QinQ interface groups- Allows rules to be written more easily).
Description - (not parsed).
Member(s) - Ranges can be specified in the inputs below. Enter a range (2-3) or individual numbers.
Click "Duplicate" as many times as needed to add new inputs.
There is no field provided in 2.3.2 to enter these
Tag(s) - text field to add tags

PPP Configuration
Link Type: PPP, PPoE, PPTP, L2TP
- *Basic options presented before selecting type:*

Link Interface(s) - Select at least two interfaces for Multilink (MLPPP).
Description- Description will appear in the "Interfaces Assign" select lists.
Username, password

Basic Options - Differences:
PPP- need phone number, enter country
PPPoE - Doesn't need those two but can optionally enter "service name" or none.
Periodic Reset timing option
PPTP, L2TP - no additional options

Advanced Options - Differences:

PPP
Access Point Name (APN)
APN number (optional) - Defaults to 1 if APN is set. Ignored if no APN is set.
SIM PIN
SIM PIN wait - Time to wait for SIM to discover network after PIN is sent to SIM (seconds).
Init string - modem initialization string. Do NOT include the "AT" string at the beginning of the command. Many modern USB 3G modems don't need an initialization string.
Connection Timeout - in seconds for conn to be established. Default is 45 sec.
Uptime logging - optional persistent logging of connection uptime. Causes cumulative uptime to be recorded and displayed on the Status->Interfaces page.

PPPoE, PPTP, L2TP All have these advanced options:
Dial On Demand - Enable mode. Do NOT enable if the link is to remain continuously connected. The interface is configured, but the actual connection is delayed until

qualifying outgoing is detected.
Idle Timeout - number of seconds idle tolerance. If the dial-on-demand option is enabled, mpd goes back into dial-on-demand mode. Otherwise, the interface is brought down and all associated routes removed.

Compression: Disable vjcomp (compression, auto-negotiated by default)
Van Jacobson TCP header compression, which saves several bytes per TCP data packet. This option is almost always required. Compression is not effective for TCP connections with enabled modern extensions like time stamping or SACK, which modify TCP options between sequential packets.

Disable tcpmssfix (enabled by default). Causes mpd to adjust incoming and outgoing TCP SYN segments so that the requested maximum segment size is not greater than the amount allowed by the interface MTU. This is necessary in many setups to avoid problems caused by routers that drop ICMP Datagram Too Big messages. Without these messages, the originating machine sends data, it passes the rogue router then hits a machine that has an MTU that is not big enough for the data. Because the IP Don't Fragment option is set, this machine sends an ICMP Datagram Too Big message back to the originator and drops the packet. The rogue router drops the ICMP message and the originator never gets to discover that it must reduce the fragment size or drop the IP Don't Fragment option from its outgoing data.

Disable shortseq (auto-negotiated by default). This option is only meaningful if multi-link PPP is negotiated. It proscribes shorter multi-link fragment headers, saving two bytes on every frame. not multi-link, it's not necessary to disable

Disable ACF compression (auto-negotiated by default) Address and control field compression. This option only applies to asynchronous link types. It saves two bytes per frame.

Disable ProtoComp (auto-negotiated by default) Protocol field compression. This option saves one byte per frame for most frames.

Definition MPD: Multi-link PPP daemon - netgraph based implementation of the multi-link PPP protocol for FreeBSD

GRE Configuration
Parent Interface - here listed are other ints and "Localhost"
GRE Remote Address Peer address where encapsulated GRE packets will be sent.
GRE tunnel local address - Local GRE tunnel endpoint.
GRE tunnel remote address - Remote GRE address endpoint.
GRE tunnel subnet - The subnet is used for determining the network that is tunnelled.
1-128 offered Not much documented on this- investigate further
Add Static Route - Check box to add an explicit static route for the remote inner tunnel address/subnet via the local tunnel address

GIF Tunnel Configuration

The same as GRE except this:
ECN friendly behavior- violates RFC2893. This should be used in mutual agreement with the peer.
Outer Source Filtering - Disable automatic filtering of the outer GIF source which ensures a match with the configured remote peer. When disabled, martian and inbound filtering is not performed which allows asymmetric routing of the outer traffic.

Interfaces - Bridges

Member Interfaces - Interfaces participating in the bridge.
Description

Just about everything is in "Advanced" options:

Cache Size - bridge address cache. The default is 2000 entries.
Cache expire time - timeout of address cache entries in seconds. If zero, address cache entries will not be expired. The default is 1200 seconds.

These port options are each simply a list of interfaces to edit:
Span Port
Add interface named as a span port on the bridge. It can't be part of the bridge member interfaces.
Edge Ports
Set interface as an edge port (connects directly to end stations and cannot create bridging loops in the network; this allows it to transition straight to forwarding.
Autoedge Ports
Allow interface to automatically detect edge status (default). Disable interfaces here.
PTP Ports
Set interface as point-to-point link; required for straight transitions to forwarding and should be enabled on a direct link to another RSTP-capable switch.
Auto PTP Ports
Autodetect point-to-point status by checking the full duplex link status (default) Like autoedge, listed are removed from default.
Sticky Ports
Mark as a "sticky" interface. Dynamically learned address entries are treated as static once entered into the cache. Sticky entries are never aged out of the cache or replaced, *even if the address is seen on a different interface.*
Private Ports
Mark an interface as a "private" interface. A private interface does not forward any traffic to any other port that is also a private interface.

Enable RSTP/STP checkbox
RSTP/STP Options
Protocol - which is used? (STP/RSTP only)
STP Interfaces (list to edit) "The if_bridge(4) driver has support for the IEEE 802.1D"
Valid time - time configuration is valid- default 20 sec - set 6-40 sec
Forward time- before interface begins forwarding packets - default 15 sec - set 4-30 sec

Hello time- only changed in legacy STP mode. The default is 2 seconds, set 1-2 sec
Priority- bridge priority - default is 32768- set 0-61440.
Hold Count - # of packets transmitted before being rate limited- default 6 - set 1-10
WAN Priority of interface to value. The default is 128, set 0-240 at increments of 16.
WAN Path cost- default calculated from link speed. To change previously selected path cost back to auto, set to 0. Set 1- 200000000.

LAGG
Link agg has no settings to speak of. You simply choose appropriate interfaces and the choose the appropriate type of aggregation:

NONE
This protocol is intended to do nothing: it disables any traffic without disabling the lagg interface itself.
LACP
Supports the IEEE 802.3ad Link Aggregation Control Protocol (LACP) and the Marker Protocol. LACP will negotiate a set of aggregable links with the peer in to one or more Link Aggregated Groups. Each LAG is composed of ports of the same speed, set to full-duplex operation. The traffic will be balanced across the ports in the LAG with the greatest total speed, in most cases there will only be one LAG which contains all ports. In the event of changes in physical connectivity, Link Aggregation will quickly converge to a new configuration.
FAILOVER
Sends and receives traffic only through the master port. If the master port becomes unavailable, the next active port is used. The first interface added is the master port; any interfaces added after that are used as failover devices.
FEC
Supports Cisco EtherChannel. This is a static setup and does not negotiate aggregation with the peer or exchange frames to monitor the link.
LOADBALANCE
Balances outgoing traffic across the active ports based on hashed protocol header information and accepts incoming traffic from any active port. This is a static setup and does not negotiate aggregation with the peer or exchange frames to monitor the link. The hash includes the Ethernet source and destination address, and, if available, the VLAN tag, and the IP source and destination address.
ROUNDROBIN
Distributes outgoing traffic using a round-robin scheduler through all active ports and accepts incoming traffic from any active port.

VPN>Tunnels>IPSEC
VPN>IPsec>Tunnels>Edit Phase 1
Three different pages of settings:
Tunnels
Mobile Clients
Pre-Shared Keys
Advanced Settings

Tunnels/ General Information
Disabled
Set this option to disable this phase1 without removing it from the list.
IKE version (v1, v2, or auto) Auto uses IKEv2 when initiator, accepts IKEv1 or IKEv2 as responder.
Internet Protocol (IPv4 or 6)
Interface- the interface for the local endpoint of this phase1 entry.
Remote Gateway- the public IP address or host name of the remote gateway.
Description

Phase 1 Proposal (Authentication)
Authentication Method - Mutual PSK (default) or Mutual RSA- Must match remote side.
Negotiation mode Main (default) or aggressive(more flexible, but less secure.)
My identifier, Peer identifier
Both of these fields offer these options:

- My (or Peer) IP address
- Any (for peer only)
- (an) IP address
- Distinguished name
- User distinguished name
- ASN.1 distinguished Name
- KeyID tag

Pre-Shared Key (field to fill or paste)
Phase 1 Proposal (Algorithms)
Encryption Algorithm: AES and 256 default
AES, AES128-GCM, AES192-GCM, AES256-GCM, Blowfish, 3DES, CAST128
Bits field: (GSM : 128, 96, 64) (Reg AES and Blowfish: 256, 192, 128) (3DES and CAST128 n/a)

Hash Algorithm: (default SHA1) - Must match remote side.
MD5, SHA1, SHA256, SHA384, SHA512, AES-XCBC

DH Group - default 2 (1024 bit) - Must match remote side.
2 (1024 bit)
5 (1536 bit)
14 (2048 bit)
15 (3072 bit)
16 (4096 bit)
17 (6144 bit)
18 (8192 bit)
19 (nist ecp256)
20 (nist ecp384)
21 (nist ecp521)
22 (1024(sub 160) bit)
23 (2048(sub 224) bit)
24 (2048(sub 256) bit)

28 (brainpool ecp256)
29 (brainpool ecp384)
30 (brainpool ecp512)

Lifetime (Seconds) default 28800
Advanced Options
Disable rekey (checkbox)
Disables renegotiation when a connection is about to expire.
Responder Only (checkbox)
Enable this option to never initiate this connection from this side, only respond to incoming requests.
NAT Traversal (default auto) - Auto or Force
Enable NAT-T (i.e. the encapsulation of ESP in UDP packets) if needed, which can help with clients that are behind restrictive firewalls.
Dead Peer Detection (checkbox) Enable DPD (checked is default)
Delay - default 10 - Delay between requesting peer acknowledgement.
Max failures - default 5 - consecutive failures allowed before disconnect.

Mobile Clients

Enable IPsec Mobile Client Support (checkbox)

IKE Extensions
Enable IPsec Mobile Client Support
Extended Authentication (Xauth)
User Authentication (Source) Local Database shown in mult-choice field
Group Authentication (Source) Choose System or None (default)

Client Configuration (mode-cfg)
Virtual Address Pool (checkbox) - Provide a virtual IP address to clients(allows specify IP/CIDR)
Virtual IPv6 Address Pool (checkbox) - Provide a virtual IPv6 address to clients (allows specify IP/CIDR)
Network List (checkbox) - Provide list of accessible networks (checking doesn't allow list edit)
Save Xauth Password (checkbox)
Allow clients to save Xauth passwords (Cisco VPN client only). NOTE: With iPhone clients, this does not work when deployed via the iPhone configuration utility, only by manual entry.
DNS Default Domain (checkbox)- Provide a default domain name to clients (gives field if checked)
Split DNS (checkbox) -Provide a list of split DNS domain names to clients. (gives field if checked) Enter a space separated list.
DNS Servers (checkbox) (gives fields if checked) - Provide a DNS server list to clients
WINS Servers (checkbox) (gives fields if checked) - Provide a WINS server list to clients
Phase2 PFS Group (checkbox) - Provide the Phase2 PFS group to clients (overrides all

mobile phase2 settings) (if checked gives pulldown with most levels listed in DH options)
Login Banner (checkbox- lets you enter banner) Provide a login banner to clients

Pre-Shared Keys
Edit Pre-Shared-Secret
Identifier (field)
This can be either an IP address, fully qualified domain name or an e-mail address. PSK for any user can be set by using an identifier of any.
Secret type PSK (default) or EAP
Pre-Shared Key (field to paste it)

Advanced Settings
IPsec Logging Controls:
For each of these: [silent | audit | control (default) | diag | raw | highest]
Daemon
SA Manager
IKE SA
IKE Child SA
Job Processing
Configuration backend
Kernel Interface
Networking
ASN encoding
Message encoding
Integrity checker
Integrity Verifier
Platform Trust Service
TLS handler
IPsec traffic
StrongSwan Lib

Advanced IPsec Settings
Configure Unique IDs as (yes-default | no | never | keep)
Whether a particular participant ID should be kept unique, with any new IKE_SA using an ID deemed to replace all old ones using that ID. Participant IDs normally are unique, so a new IKE_SA using the same ID is almost invariably intended to replace an old one. The difference between no and never is that the old IKE_SAs will be replaced when receiving an INITIAL_CONTACT notify if the option is no but will ignore these notifies if never is configured. The daemon also accepts the value keep to reject new IKE_SA setups and keep the duplicate established earlier. Defaults to Yes.

The rest of these are checkboxes, all off except "Auto-exclude LAN address"
IP Compression- IPComp compression of content is proposed on the connection.
Strict interface binding- Enable strongSwan's interfaces_use option to bind specific interfaces only. This option is known to break IPsec with dynamic IP interfaces. This is not recommended at this time.

Unencrypted payloads in IKEv1 Main Mode-
Accept unencrypted ID and HASH payloads in IKEv1 Main Mode Some implementations send the third Main Mode message unencrypted, probably to find the PSKs for the specified ID for authentication. This is very similar to Aggressive Mode, and has the same security implications: A passive attacker can sniff the negotiated Identity, and start brute forcing the PSK using the HASH payload. It is recommended to keep this option to no, unless the exact implications are known and compatibility is required for such devices (for example, some SonicWall boxes).

Enable Maximum MSS- Enable MSS clamping on VPN traffic
Enable Cisco Extensions (Unity Plugin) support Split-Include, Split-Exclude and Split-Dns.
Strict CRL Checking- Check this to require availability of a fresh CRL for peer authentication based on RSA signatures to succeed.

Make before Break
Initiate IKEv2 reauthentication with a make-before-break Instead of a break-before-make scheme. Make-before-break uses overlapping IKE and CHILD_SA during reauthentication by first recreating all new SAs before deleting the old ones. This behavior can be beneficial to avoid connectivity gaps during reauthentication, but requires support for overlapping SAs by the peer.

Auto-exclude LAN address- default is checked
Enable bypass for LAN interface IP Exclude traffic from LAN subnet to LAN IP address from IPsec.

VPN - L2TP
Configuration
Enable LT2P server *(nothing below is going to show up until you check this box!)*
"Don't forget to add a firewall rule to permit traffic from L2TP clients."

Interface
Server address- the IP address the L2TP server should give to clients for use as their "gateway".
Typically an unused IP just outside of the client range.
NOTE: This should NOT be set to any IP address currently in use on this firewall.

Remote address range- (IP+ subnet 1-128) Specify the starting address for the client IP address subnet.
Number of L2TP users (1-255)
Secret (optional secret shared between peers required on some devices/setups.
Authentication type (PAP | CHAP | MS-CHAPv2)
Primary L2TP DNS server
Secondary L2TP DNS server

<u>RADIUS</u>

Enable *(nothing below is going to show up until you check this box!)*
Use a RADIUS server for authentication When set, all users will be authenticated using the RADIUS server specified below. The local user database will not be used.
Accounting
Enable RADIUS accounting Sends accounting packets to the RADIUS server.
Server
Enter the IP address of the RADIUS server.
Secret
Enter the shared secret that will be used to authenticate to the RADIUS server.
Confirm
RADIUS issued IPs
Issue IP Addresses via RADIUS server.

There is a Users page, and all it asks is username, password, and IP address

OpenVPN
Has 4 pages for configuration:
Servers, Clients, Client Specific Overrides, Wizards

Disable this server (checkbox) disable this server without removing it from the list.
Server mode -
Peer to Peer: (SSL/TSL or Shared Key), Remote Access (SSL/TSL, User Auth, SSL/TSL+User Auth)

Section for Server mode: Peer to Peer (SSL/TSL or Shared Key), and Remote Access (SSL/TSL)
User Auth adds Backend for authentication: (default is Local Database)
Protocol (UDP | TCP | UDP6 | TCP6]
Device mode - default tun or tap
Interface: in addition to standard, you will see localhost and any
Local port (default 1194)
Description

Cryptographic Settings
Shared key will won't ask about certificates, and has same crypto and hash options
User Auth will add "Strict User-CN Matching":
"Enforce match When authenticating users, enforce a match between the common name of the client certificate and the username given at login. "

(this is for SSL/TSL modes)
TLS authentication
Enable authentication of TLS packets (checkbox - default checked)
Automatically generate a shared TLS authentication key (checkbox - default checked)
Peer Certificate Authority (appears in place of CRLs when you first load the "Add Server" window)
Peer Certificate Revocation list- "No CRLs defined. Create at: System > Cert. Manager
Server certificate (chose one from list)

DH Parameter length (bits) [1024 (default), 2048, 3072, 4096, 7680, 8192, 15360, 16384]
Encryption Algorithm - HUGE list to chose from:

- AES-128-CBC | CFB | CFB1 | CFB8 | OFB (128-bit)
- AES-192-CBC | CFB | CFB1 | CFB8 | OFB (192-bit)
- AES-256-CBC | CFB | CFB1 | CFB8 | OFB (256-bit)
- BF-CBC | CFB | OFB (128-bit)
- CAMELLIA-128-CBC | CFB | CFB1 | CFB8 | OFB (128-bit)
- CAMELLIA-192-CBC | CFB | CFB1 | CFB8 | OFB (192-bit)
- CAMELLIA-256-CBC | CFB | CFB1 | CFB8 | OFB (256-bit)
- CAST5-CBC | CFB | OFB (128-bit)
- DES-CBC | CFB | CFB1 | CFB8 (64-bit)
- DES-EDE-CBC | CFB | OFB (128-bit)
- DES-EDE3-CBC | CFB | CFB1 | CFB8 | OFB (192-bit)
- DES-OFB (64-bit)
- DESX-CBC (192-bit)
- IDEA-CBC | CFB | OFB (128-bit)
- RC2-40-CBC (40-bit)
- RC2-64-CBC (64-bit)
- RC2-CBC | CFB | OFB (128-bit)
- RC5-CBC | CFB | OFB (128-bit)
- SEED-CBC | CFB | OFB (128-bit)
- None (No Encryption)

Auth digest algorithm (SHA1 is the default for OpenVPN.

- DSA | DSA-SHA | DSA-SHA1 | SHA1-old (160-bit)
- MD4 (128-bit)
- MD5 (128-bit)
- MDC2 (128-bit)
- RIPEMD160 (160-bit)
- RSA-MD4 | MD5 | MDC2 (128-bit)
- RSA-RIPEMD160 | SHA | SHA1 | SHA1-2 (160-bit)
- RSA-SHA224 (224-bit)
- RSA-SHA256 (256-bit)
- RSA-SHA384 (384-bit)
- RSA-SHA512 (512-bit)
- SHA (160-bit)
- SHA1 (160-bit)
- SHA224 (224-bit)
- SHA256 (256-bit)
- SHA384 (384-bit)
- SHA512 (512-bit)
- ecdsa-with-SHA1 (160-bit)
- whirlpool (512-bit)
- None (No Authentication)

Hardware Crypto

No Hardware Crypto Acceleration
BSD cryptodev engine - RSA, DSA, DH
Intel RDRAND engine - RAND

Certificate Depth

Do Not Check
One (Client+Server)
Two (Client+Intermediate+Server)
Three (Client+2xIntermediate+Server)
Four (Client+3xIntermediate+Server)
Five (Client+4xIntermediate+Server)

When a certificate-based client logs in, do not accept certificates below this depth. Useful for denying certificates made with intermediate CAs generated from the same CA as the server.

Tunnel Settings

IPv4 Tunnel Network

Virtual network used for private communications between this server and client hosts expressed using CIDR (e.g. 10.0.8.0/24). The first network address will be assigned to the server virtual interface. The remaining network addresses can optionally be assigned to connecting clients (see Address Pool).

IPv6 Tunnel Network

Same as above for IPv6, expressed using CIDR (e.g. fe80::/64). .

Redirect Gateway

Force all client generated traffic through the tunnel.

IPv4 Local network(s)

IPv4 networks that will be accessible from the remote endpoint. Expressed as a comma-separated list of one or more CIDR ranges. This may be left blank if not adding a route to the local network through this tunnel on the remote machine. This is generally set to the LAN network.

IPv6 Local network(s)

Same as above for IPv6 - use IP/PREFIX

IPv4 Remote network(s)----- Peer to Peer only

IPv4 networks that will be routed through the tunnel, so that a site-to-site VPN can be established without manually changing the routing tables. Expressed as a comma-separated list of one or more CIDR ranges. If this is a site-to-site VPN, enter the remote LAN/s here. May be left blank for non site-to-site VPN.

IPv6 Remote network(s)----- Peer to Peer only

Same as abovefor IPv6 using IP/PREFIX

Concurrent connections - max # of clients allowed to concurrently connect to this server.

Compression - no preference, disabled, enabled with, or enabled w/o adaptive
LZO algorithm on tunnel packets. Adaptive compression will dynamically disable compression for a period of time if OpenVPN detects that the data in the packets is not being compressed efficiently.

Type-of-Service
Set the TOS IP header value of tunnel packets to match the encapsulated packet value.

Inter-client communication (Remote Access setting only)
Allow communication between clients connected to this server

Duplicate Connection
Allow multiple concurrent connections from clients using the same Common Name.
(This is not generally recommended, but may be needed for some scenarios.)

Disable IPv6 - Don't forward IPv6 traffic.

Client Settings (none for shared key)
Dynamic IP (default not checked)
Allow connected clients to retain their connections if their IP address changes.

Address Pool (default checked)
Provide a virtual adapter IP address to clients (see Tunnel Network).

Topology (tunnel mode only)
    Subnet -- One IP address per client in a common subnet
    net30 -- Isolated /30 network per client
Specifies the method used to supply a virtual adapter IP address to clients when using TUN mode on IPv4. Some clients may require this be set to "subnet" even for IPv6, such as OpenVPN Connect (iOS/Android). Older versions of OpenVPN (before 2.0.9) or clients such as Yealink phones may require "net30".

Advanced Client Settings - Remote Access-only
DNS Default Domain
Provide a default domain name to clients
DNS Server enable
Provide a DNS server list to clients
Force DNS cache update
Run "net stop dnscache", "net start dnscache", "ipconfig /flushdns" and "ipconfig /registerdns" on connection initiation. This is known to kick Windows into recognizing pushed DNS servers.
NTP Server enable
Provide an NTP server list to clients
NetBIOS enable
Enable NetBIOS over TCP/IP If this option is not set, all NetBIOS-over-TCP/IP options (including WINS) will be disabled.
Enable custom port

Use a different management port for clients.

Advanced Configuration:

Custom options
Enter any additional options to add to the OpenVPN server configuration here, separated by semicolon. EXAMPLE: push "route 10.0.0.0 255.255.255.0"

Verbosity level
Each level shows all info from the previous levels. Level 3 is recommended for a good summary of what's happening without being swamped by output.

None: Only fatal errors
Default through 4: Normal usage range
5: Output R and W characters to the console for each packet read and write. Uppercase is used for TCP/UDP packets and lowercase is used for TUN/TAP packets.
6-11: Debug info range

Add client has many similar settings. Can only do Peer to peer.

Don't pull routes (default is disabled)
Bars the server from adding routes to the client's routing table This option still allows the server to set the TCP/IP properties of the client's TUN/TAP interface.

Don't add/remove routes (default is disabled)
Don't add or remove routes automatically Pass routes to --route-upscript using environmental variables.

Adds these fields:
Proxy host or address
Proxy port
Proxy Auth. - Extra options - none, basic, or ntlm
Server hostname resolution
Infinitely resolve server (default is disabled)
Continuously attempt to resolve the server host name. Useful when communicating with a server that is not permanently connected to the Internet.

Limit outgoing bandwidth (default is the range below, and greyed out on my screen)
Maximum outgoing bandwidth for this tunnel. Leave empty for no limit. The input value has to be something between 100 bytes/sec and 100 Mbytes/sec (entered as bytes per second).

There is also a Client-Overrides page, which allows replacing some server settings

---

OpenVPN wizard
Question 1 Local User Access (default), LDAP, or RADIUS
Pages 2- create a CA certificate
Key size, lifetime Country code, State/Province, City, Org and email
Page 3 - server certificate
Page 4 Shared key TLS and Name server info

CVSup (5999)
DNS (53)
FTP (21)
HBCI (3000)
HTTP (80)
HTTPS (443)
ICQ (5190)
IDENT/AUTH (113)
IMAP (143)
IMAP/S (993)
IPsec NAT-T (4500)
ISAKMP (500)
L2TP (1701)
LDAP (389)
MMS/TCP (1755)
MMS/UDP (7000)
MS DS (445)
MS RDP (3389)
MS WINS (1512)
MSN (1863)
NNTP (119)
NTP (123)
NetBIOS-DGM (138)
NetBIOS-NS (137)
NetBIOS-SSN (139)
OpenVPN (1194)
POP3 (110)
POP3/S (995)
PPTP (1723)
RADIUS (1812)
RADIUS accounting (1813)
RTP (5004)
SIP (5060)

SMTP (25)
SMTP/S (465)
SNMP (161)
SNMP-Trap (162)
SSH (22)
STUN (3478)
SUBMISSION (587)
Teredo (3544)
Telnet (23)
TFTP (69)
VNC (5900)

Snort Detection Performance Settings
Search Method
AC-BNFA
AC-SPLIT
LOWMEM
AC-STD
AC
AC-NQ
AC-BNFA-NQ
LOWMEM-NQ
AC-BANDED
AC-SPARSEBANDS
ACS

ac Aho-Corasick Full (high memory, best performance)
ac-std Aho-Corasick Standard (moderate memory, high performance)
ac-bnfa Aho-Corasick NFA (low memory, high performance)
acs Aho-Corasick Sparse (small memory, moderate performance)
ac-banded Aho-Corasick Banded (small memory, moderate performance)
ac-sparsebands Aho-Corasick Sparse-Banded (small memory, high performance)
lowmem Low Memory Keyword Trie (small memory, low performance)

Snort Interfaces
Global Settings
Updates
Alerts
Blocked
Pass Lists
Suppress
IP Lists
SID Mgmt
Log Mgmt
Sync

Iface Settings
Iface Categories
Iface Rules
Iface Variables
Iface Preprocs
Iface Barnyard2
Iface IP Rep
Iface Logs

Sample crash dump:

Crash report begins. Anonymous machine information:

i386
10.3-RELEASE-p5
FreeBSD 10.3-RELEASE-p5 #0 7307492(RELENG_2_3_2): Tue Jul 19 13:53:47 CDT 2016 root@ce23-i386-builder:/builder/pfsense-232/tmp/obj/builder/pfsense-232/tmp/FreeBSD-src/sys/pfSense

Crash report details:

Filename: /var/crash/bounds
1

Filename: /var/crash/info.0
Dump header from device /dev/label/swap0
Architecture: i386
Architecture Version: 1
Dump Length: 66560B (0 MB)
Blocksize: 512
Dumptime: Wed Aug 17 04:30:22 2016
Hostname:
Magic: FreeBSD Text Dump
Version String: FreeBSD 10.3-RELEASE-p5 #0 7307492(RELENG_2_3_2): Tue Jul 19 13:53:47 CDT 2016
root@ce23-i386-builder:/builder/pfsense-232/tmp/obj/builder/pfsense-232/tmp/FreeBSD-src/sys/pfSense
Panic String: ffs_valloc: dup alloc
Dump Parity: 1639756106
Bounds: 0
Dump Status: good

Filename: /var/crash/info.last
Dump header from device /dev/label/swap0

Architecture: i386
Architecture Version: 1
Dump Length: 66560B (0 MB)
Blocksize: 512
Dumptime: Wed Aug 17 04:30:22 2016
Hostname:
Magic: FreeBSD Text Dump
Version String: FreeBSD 10.3-RELEASE-p5 #0 7307492(RELENG_2_3_2): Tue Jul 19 13:53:47 CDT 2016
root@ce23-i386-builder:/builder/pfsense-232/tmp/obj/builder/pfsense-232/tmp/FreeBSD-src/sys/pfSense
Panic String: ffs_valloc: dup alloc
Dump Parity: 1639756106
Bounds: 0
Dump Status: good

Filename: /var/crash/minfree
2048

Filename: /var/crash/textdump.tar.0
ddb.txt0600001365711275476413 6 7133 ustarrootwheeldb:0:kdb.enter.default> run lockinfo
db:1:lockinfo> show locks
No such command
db:1:locks> show alllocks
No such command
db:1:alllocks> show lockedvnods
Locked vnodes
db:0:kdb.enter.default> show pcpu
cpuid = 0
dynamic pcpu = 0x56f000
curthread = 0xc6f4cc80: pid 149 "php-cgi"
curpcb = 0xe9af6c00
fpcurthread = none
idlethread = 0xc6716000: tid 100003 "idle: cpu0"
APIC ID = 0
currentldt = 0x50
db:0:kdb.enter.default> bt
Tracing pid 149 tid 100069 td 0xc6f4cc80
kdb_enter(c147c5b6,c147c5b6,c14afaab,e9af667c,0,...) at kdb_enter+0x3d/frame 0xe9af6628
vpanic(c14afaab,e9af667c,e9af667c,e9af66f8,c10007b1,...) at vpanic+0x13b/frame 0xe9af665c
panic(c14afaab,81b6,11,c6dc00d4,8180,...) at panic+0x1b/frame 0xe9af6670
ffs_valloc(c6edfd50,8180,c6dbcc00,e9af673c,0,...) at ffs_valloc+0x961/frame 0xe9af66f8
ufs_makeinode(e9af6a50,e9af6a64,c6edfd50,e9af6a08,e9af68c8,...) at

```
ufs_makeinode+0x73/frame 0xe9af6878
ufs_create(e9af6970,2,c6f4cc80,c1eb5990,c6edfd50,...) at ufs_create+0x30/frame
0xe9af689c
VOP_CREATE_APV(c1ea0728,e9af6970,2,c6edfd50,0,...) at
VOP_CREATE_APV+0x95/frame 0xe9af68c8
vn_open_cred(e9af6a08,e9af6a94,180,0,c6dbcc00,c6dde4d0) at vn_open_cred+0x2ae/
frame 0xe9af6998
vn_open(e9af6a08,e9af6a94,180,c6dde4d0,287e5e28,...) at vn_open+0x3d/frame
0xe9af69c0
kern_openat(c6f4cc80,ffffff9c,287e5e28,0,a02,180) at kern_openat+0x310/frame
0xe9af6ab4
sys_open(c6f4cc80,e9af6b68,c6f48c20,c0d1bba2,c6f4cc80,...) at sys_open+0x39/
frame 0xe9af6ad8
syscall(e9af6ba8) at syscall+0x5c9/frame 0xe9af6b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9af6b9c
--- syscall (5, FreeBSD ELF32, sys_open), eip = 0x287447af, esp = 0xbfbfe06c, ebp =
0xbfbfe514 ---
db:0:kdb.enter.default>  ps
  pid  ppid  pgrp   uid   state   wmesg     wchan    cmd
  150   148    21     0  S+      piperd   0xc6dee000 grep
  149   148    21     0  R+      CPU 0                php-cgi
  148   136    21     0  S+      wait     0xc6f49610 sh
  136    21    21     0  S+      piperd   0xc6dee198 sh
   55     0     0     0  DL      mdwait   0xc6d7d800 [md0]
   21     1    21     0  Ss+     pause    0xc6d8a97c sh
   20     0     0     0  DL      vlruwt   0xc6d8ac20 [vnlru]
   19     0     0     0  DL      syncer   0xc1f99fc4 [syncer]
   18     0     0     0  DL      (threaded)          [bufdaemon]
100061                  D       sdflush  0xc6914284 [/ worker]
100054                  D       psleep   0xc1f99704 [bufdaemon]
   17     0     0     0  DL      pgzero   0xc1fa41a8 [pagezero]
    9     0     0     0  DL      pollid   0xc1f797f0 [idlepoll]
    8     0     0     0  DL      psleep   0xc1fa3ec4 [vmdaemon]
    7     0     0     0  DL      (threaded)          [pagedaemon]
100057                  D       umarcl   0xc1fa3aa4 [uma]
100050                  D       psleep   0xc2027604 [pagedaemon]
    6     0     0     0  DL      waiting_ 0xc2021f74 [sctp_iterator]
    5     0     0     0  DL      pftm     0xc0f94ba0 [pf purge]
   16     0     0     0  DL      (threaded)          [usb]
100041                  D       -        0xc6893d64 [usbus1]
100040                  D       -        0xc6893d34 [usbus1]
100039                  D       -        0xc6893d04 [usbus1]
100038                  D       -        0xc6893cd4 [usbus1]
100037                  D       -        0xc6893ca4 [usbus1]
100033                  D       -        0xc67fcb8c [usbus0]
100032                  D       -        0xc67fcb5c [usbus0]
100031                  D       -        0xc67fcb2c [usbus0]
```

```
100030                D      -        0xc67fcafc [usbus0]
100029                D      -        0xc67fcacc [usbus0]
    4     0     0     0  DL     (threaded)          [cam]
100049                D      -        0xc1eb8eac [scanner]
100017                D      -        0xc1eb9000 [doneq0]
   15     0     0     0  DL     -        0xc1ed60c8 [rand_harvestq]
    3     0     0     0  DL     crypto_r 0xc1fa2e28 [crypto returns]
    2     0     0     0  DL     crypto_w 0xc1fa2d68 [crypto]
   14     0     0     0  DL     (threaded)          [geom]
100010                D      -        0xc201cfd8 [g_down]
100009                D      -        0xc201cfd4 [g_up]
100008                D      -        0xc201cfd0 [g_event]
   13     0     0     0  DL     sleep    0xc1e9081c [ng_queue0]
   12     0     0     0  WL     (threaded)          [intr]
100047                I                        [swi1: pfsync]
100045                I                        [swi1: pf send]
100043                I                        [irq12: psm0]
100042                I                        [irq1: atkbd0]
100036                I                        [irq10: em0 ehci0]
100028                I                        [irq11: pcm0 em2+]
100026                I                        [irq15: ata1]
100025                I                        [irq14: ata0]
100024                I                        [swi6: Giant taskq]
100022                I                        [swi6: task queue]
100016                I                        [swi5: fast taskq]
100006                I                        [swi4: clock]
100005                I                        [swi1: netisr 0]
100004                I                        [swi3: vm]
   11     0     0     0  RL                        [idle: cpu0]
    1     0     1     0  SLs    wait     0xc670f308 [init]
   10     0     0     0  DL     audit_wo 0xc2025edc [audit]
    0     0     0     0  DLs    (threaded)          [kernel]
100048                D      -        0xc6712980 [CAM taskq]
100035                D      -        0xc68d0480 [em2 taskq]
100034                D      -        0xc68d0700 [em1 taskq]
100027                D      -        0xc6883300 [em0 taskq]
100023                D      -        0xc6712400 [ffs_trim taskq]
100021                D      -        0xc6712680 [kqueue taskq]
100020                D      -        0xc6712700 [acpi_task_2]
100019                D      -        0xc6712700 [acpi_task_1]
100018                D      -        0xc6712700 [acpi_task_0]
100015                D      -        0xc6712b00 [thread taskq]
100011                D      -        0xc6713e80 [firmware taskq]
100000                D      swapin   0xc201d054 [swapper]
db:0:kdb.enter.default>  alltrace

Tracing command grep pid 150 tid 100067 td 0xc6f4d320
```

sched_switch(c6f4d320,0,104,ac,98,...) at sched_switch+0x2d2/frame 0xe9af089c
mi_switch(104,0,c6f4d320,e9af094c,c0cf40f2,c6f4d320) at mi_switch+0x122/frame 0xe9af08d4
sleepq_switch(c6f4d320,0,c1480b9a,1a8,e9af0960,...) at sleepq_switch+0x15b/frame 0xe9af08fc
sleepq_catch_signals(0,c1480b9a,154,0,0,...) at sleepq_catch_signals+0x5be/frame 0xe9af0950
sleepq_wait_sig(c6dee000,5c,c1481733,100,0,...) at sleepq_wait_sig+0x14/frame 0xe9af097c
_sleep(c6dee000,c6dee180,15c,c1481733,0,...) at _sleep+0x29b/frame 0xe9af09c4
pipe_read(c6dde658,e9af0aa8,c6dbcc00,0,c6f4d320,...) at pipe_read+0x31a/frame 0xe9af0a08
dofileread(c6dde658,e9af0aa8,ffffffff,ffffffff,0,...) at dofileread+0x9e/frame 0xe9af0a40
kern_readv(c6f4d320,0,e9af0aa8,e9af0ad8,2881c000,...) at kern_readv+0x96/frame 0xe9af0a88
sys_read(c6f4d320,e9af0b68,c6f49308,c6f4d320,c6f49308,...) at sys_read+0x5c/frame 0xe9af0ad8
syscall(e9af0ba8) at syscall+0x5c9/frame 0xe9af0b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9af0b9c
--- syscall (3, FreeBSD ELF32, sys_read), eip = 0x281de85f, esp = 0xbfbfec18, ebp = 0xbfbfec60 ---

Tracing command php-cgi pid 149 tid 100069 td 0xc6f4cc80
kdb_enter(c147c5b6,c147c5b6,c14afaab,e9af667c,0,...) at kdb_enter+0x3d/frame 0xe9af6628
vpanic(c14afaab,e9af667c,e9af667c,e9af66f8,c10007b1,...) at vpanic+0x13b/frame 0xe9af665c
panic(c14afaab,81b6,11,c6dc00d4,8180,...) at panic+0x1b/frame 0xe9af6670
ffs_valloc(c6edfd50,8180,c6dbcc00,e9af673c,0,...) at ffs_valloc+0x961/frame 0xe9af66f8
ufs_makeinode(e9af6a50,e9af6a64,c6edfd50,e9af6a08,e9af68c8,...) at ufs_makeinode+0x73/frame 0xe9af6878
ufs_create(e9af6970,2,c6f4cc80,c1eb5990,c6edfd50,...) at ufs_create+0x30/frame 0xe9af689c
VOP_CREATE_APV(c1ea0728,e9af6970,2,c6edfd50,0,...) at VOP_CREATE_APV+0x95/frame 0xe9af68c8
vn_open_cred(e9af6a08,e9af6a94,180,0,c6dbcc00,c6dde4d0) at vn_open_cred+0x2ae/frame 0xe9af6998
vn_open(e9af6a08,e9af6a94,180,c6dde4d0,287e5e28,...) at vn_open+0x3d/frame 0xe9af69c0
kern_openat(c6f4cc80,ffffff9c,287e5e28,0,a02,180) at kern_openat+0x310/frame 0xe9af6ab4
sys_open(c6f4cc80,e9af6b68,c6f48c20,c0d1bba2,c6f4cc80,...) at sys_open+0x39/frame 0xe9af6ad8
syscall(e9af6ba8) at syscall+0x5c9/frame 0xe9af6b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9af6b9c
--- syscall (5, FreeBSD ELF32, sys_open), eip = 0x287447af, esp = 0xbfbfe06c, ebp =

0xbfbfe514 ---

Tracing command sh pid 148 tid 100066 td 0xc6f4d640
sched_switch(c6f4d640,0,104,ac,98,...) at sched_switch+0x2d2/frame 0xe9aed820
mi_switch(104,0,e9aed8d0,c0cf40f2,c1f86e80,c6f4d640) at mi_switch+0x122/frame 0xe9aed858
sleepq_switch(c6f4d640,0,c1480b9a,1a8,0,...) at sleepq_switch+0x15b/frame 0xe9aed880
sleepq_catch_signals(0,c1480b9a,154,0,3457e5ce,...) at sleepq_catch_signals+0x5be/frame 0xe9aed8d4
sleepq_wait_sig(c6f49610,6c,c1481bd3,100,0,...) at sleepq_wait_sig+0x14/frame 0xe9aed900
_sleep(c6f49610,c6f496a8,16c,c1481bd3,0,...) at _sleep+0x29b/frame 0xe9aed948
kern_wait6(c6f4d640,7,0,0,e9aed9e4,...) at kern_wait6+0x71a/frame 0xe9aed9b8
sys_wait4(c6f4d640,e9aedb68,c6f49610,c6f4d640,c6f49610,...) at sys_wait4+0x94/frame 0xe9aedad8
syscall(e9aedba8) at syscall+0x5c9/frame 0xe9aedb9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9aedb9c
--- syscall (7, FreeBSD ELF32, sys_wait4), eip = 0x281acf43, esp = 0xbfbfe918, ebp = 0xbfbfe930 ---

Tracing command sh pid 136 tid 100062 td 0xc6d96000
sched_switch(c6d96000,0,104,ac,98,...) at sched_switch+0x2d2/frame 0xe9ade89c
mi_switch(104,0,c6d96000,e9ade94c,c0cf40f2,c6d96000) at mi_switch+0x122/frame 0xe9ade8d4
sleepq_switch(c6d96000,0,c1480b9a,1a8,c27a284c,...) at sleepq_switch+0x15b/frame 0xe9ade8fc
sleepq_catch_signals(0,c1480b9a,154,0,0,...) at sleepq_catch_signals+0x5be/frame 0xe9ade950
sleepq_wait_sig(c6dee198,5c,c1481733,100,0,...) at sleepq_wait_sig+0x14/frame 0xe9ade97c
_sleep(c6dee198,c6dee318,15c,c1481733,0,...) at _sleep+0x29b/frame 0xe9ade9c4
pipe_read(c6dde5b0,e9adeaa8,c6dbcc00,0,c6d96000,...) at pipe_read+0x31a/frame 0xe9adea08
dofileread(c6dde5b0,e9adeaa8,ffffffff,ffffffff,0,...) at dofileread+0x9e/frame 0xe9adea40
kern_readv(c6d96000,3,e9adeaa8,e9adead8,bfbfeb60,...) at kern_readv+0x96/frame 0xe9adea88
sys_read(c6d96000,e9adeb68,c6d8a000,0,0,...) at sys_read+0x5c/frame 0xe9adead8
syscall(e9adeba8) at syscall+0x5c9/frame 0xe9adeb9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9adeb9c
--- syscall (3, FreeBSD ELF32, sys_read), eip = 0x2820f85f, esp = 0xbfbfeacc, ebp = 0xbfbfebf4 ---

Tracing command md0 pid 55 tid 100064 td 0xc690c960
sched_switch(c690c960,0,104,e9ae6a70,0,...) at sched_switch+0x2d2/frame 0xe9ae6a48
mi_switch(104,0,da8c2370,c6e02100,c690c960,c6d7d800) at mi_switch+0x122/frame

0xe9ae6a80
sleepq_switch(c690c960,0,c1480b9a,269,c6dbc3d8,...) at sleepq_switch+0x15b/frame 0xe9ae6aa8
sleepq_wait(c6d7d800,5c,c1401bd9,0,0,...) at sleepq_wait+0x3f/frame 0xe9ae6ad4
_sleep(c6d7d800,c6d7d820,25c,c1401bd9,0,...) at _sleep+0x2ae/frame 0xe9ae6b1c
md_kthread(c6d7d800,e9ae6ba8,c1eb1abc,2880a4a0,e9ae6b98,...) at md_kthread+0xd7/frame 0xe9ae6b64
fork_exit(c08726b0,c6d7d800,e9ae6ba8) at fork_exit+0xa3/frame 0xe9ae6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xe9ae6b94
--- trap 0, eip = 0, esp = 0xe9ae6be0, ebp = 0 ---

Tracing command sh pid 21 tid 100058 td 0xc6d96c80
sched_switch(c6d96c80,0,104,ac,98,...) at sched_switch+0x2d2/frame 0xe9ad093c
mi_switch(104,0,c6d96c80,e9ad09ec,c0cf40f2,c6d96c80) at mi_switch+0x122/frame 0xe9ad0974
sleepq_switch(c6d96c80,0,c1480b9a,1a8,0,...) at sleepq_switch+0x15b/frame 0xe9ad099c
sleepq_catch_signals(0,c1480b9a,154,c6d96c80,e9ad0a64,...) at sleepq_catch_signals+0x5be/frame 0xe9ad09f0
sleepq_wait_sig(c6d8a97c,74,c1418125,100,0,...) at sleepq_wait_sig+0x14/frame 0xe9ad0a1c
_sleep(c6d8a97c,c6d8a9b0,174,c1418125,0,...) at _sleep+0x29b/frame 0xe9ad0a64
kern_sigsuspend(c6d96c80,0,0,0,0,...) at kern_sigsuspend+0x137/frame 0xe9ad0aa8
sys_sigsuspend(c6d96c80,e9ad0b68,c6d8a918,0,0,...) at sys_sigsuspend+0x58/frame 0xe9ad0ad8
syscall(e9ad0ba8) at syscall+0x5c9/frame 0xe9ad0b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9ad0b9c
--- syscall (4, FreeBSD ELF32, sys_write), eip = 0x281acf83, esp = 0xbfbfec0c, ebp = 0xbfbfec98 ---

Tracing command vnlru pid 20 tid 100056 td 0xc6d97320
sched_switch(c6d97320,0,104,0,c201e5a4,...) at sched_switch+0x2d2/frame 0xdb6a2a1c
mi_switch(104,0,101,c6d97320,db6a2aa8,c6d8ac20) at mi_switch+0x122/frame 0xdb6a2a54
sleepq_switch(c6d97320,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdb6a2a7c
sleepq_timedwait(c6d8ac20,60,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdb6a2aa8
_sleep(c6d8ac20,c1f99f7c,260,c1487ef8,ffffffed8,...) at _sleep+0x282/frame 0xdb6a2af0
vnlru_proc(0,db6a2ba8,0,0,0,...) at vnlru_proc+0xcf/frame 0xdb6a2b64
fork_exit(c0dc7f10,0,db6a2ba8) at fork_exit+0xa3/frame 0xdb6a2b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdb6a2b94
--- trap 0, eip = 0, esp = 0xdb6a2be0, ebp = 0 ---

Tracing command syncer pid 19 tid 100055 td 0xc6d97640
sched_switch(c6d97640,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xdb69fa28

mi_switch(104,0,101,c6d97640,db69fab4,c1f99fc4) at mi_switch+0x122/frame 0xdb69fa60
sleepq_switch(c6d97640,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdb69fa88
sleepq_timedwait(c1f99fc4,0,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdb69fab4
_cv_timedwait_sbt(c1f99fc4,c1f99fa4,fffffed8,0,0,...) at _cv_timedwait_sbt+0x1b7/frame 0xdb69fae8
sched_sync(0,db69fba8,0,0,0,...) at sched_sync+0x7a1/frame 0xdb69fb64
fork_exit(c0dc75d0,0,db69fba8) at fork_exit+0xa3/frame 0xdb69fb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdb69fb94
--- trap 0, eip = 0, esp = 0xdb69fbe0, ebp = 0 ---

Tracing command bufdaemon pid 18 tid 100061 td 0xc6d96320
sched_switch(c6d96320,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xe9adba50
mi_switch(104,0,101,c6d96320,e9adbadc,c6914284) at mi_switch+0x122/frame 0xe9adba88
sleepq_switch(c6d96320,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xe9adbab0
sleepq_timedwait(c6914284,54,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xe9adbadc
_sleep(c6914284,c6914200,54,c14b2d06,7fffff6c,...) at _sleep+0x282/frame 0xe9adbb24
softdep_flush(c6db4d20,e9adbba8,c10178f0,0,e9adbb90,...) at softdep_flush+0x1fa/frame 0xe9adbb64
fork_exit(c10178f0,c6db4d20,e9adbba8) at fork_exit+0xa3/frame 0xe9adbb94
fork_trampoline() at fork_trampoline+0x8/frame 0xe9adbb94
--- trap 0, eip = 0, esp = 0xe9adbbe0, ebp = 0 ---

Tracing command bufdaemon pid 18 tid 100054 td 0xc6d97960
sched_switch(c6d97960,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xdb69ca58
mi_switch(104,0,101,c6d97960,db69cae4,c1f99704) at mi_switch+0x122/frame 0xdb69ca90
sleepq_switch(c6d97960,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdb69cab8
sleepq_timedwait(c1f99704,54,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdb69cae4
_sleep(c1f99704,c1f99680,54,c148589d,fffffed8,...) at _sleep+0x282/frame 0xdb69cb2c
buf_daemon(0,db69cba8,0,0,0,...) at buf_daemon+0xac/frame 0xdb69cb64
fork_exit(c0dab150,0,db69cba8) at fork_exit+0xa3/frame 0xdb69cb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdb69cb94
--- trap 0, eip = 0, esp = 0xdb69cbe0, ebp = 0 ---

Tracing command pagezero pid 17 tid 100053 td 0xc6d97c80
sched_switch(c6d97c80,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xdb699a60
mi_switch(104,0,101,c6d97c80,db699aec,c1fa41a8) at mi_switch+0x122/frame 0xdb699a98

sleepq_switch(c6d97c80,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdb699ac0
sleepq_timedwait(c1fa41a8,0,12b,0,0,...) at sleepq_timedwait+0x3f/frame 0xdb699aec
_sleep(c1fa41a8,c2027580,0,c14b71b2,fffea520,...) at _sleep+0x282/frame 0xdb699b34
vm_pagezero(0,db699ba8,0,0,0,...) at vm_pagezero+0xd2/frame 0xdb699b64
fork_exit(c107ad10,0,db699ba8) at fork_exit+0xa3/frame 0xdb699b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdb699b94
--- trap 0, eip = 0, esp = 0xdb699be0, ebp = 0 ---

Tracing command idlepoll pid 9 tid 100052 td 0xc68b9c80
sched_switch(c68b9c80,0,104,0,c201e5a4,...) at sched_switch+0x2d2/frame 0xdabffa4c
mi_switch(104,0,101,c68b9c80,dabffad8,c1f797f0) at mi_switch+0x122/frame 0xdabffa84
sleepq_switch(c68b9c80,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdabffaac
sleepq_timedwait(c1f797f0,0,2,0,0,...) at sleepq_timedwait+0x3f/frame 0xdabffad8
_sleep(c1f797f0,0,0,c147ae9d,fffffc88,...) at _sleep+0x282/frame 0xdabffb20
poll_idle(0,dabffba8,0,0,0,...) at poll_idle+0x125/frame 0xdabffb64
fork_exit(c0cf7de0,0,dabffba8) at fork_exit+0xa3/frame 0xdabffb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabffb94
--- trap 0, eip = 0, esp = 0xdabffbe0, ebp = 0 ---

Tracing command vmdaemon pid 8 tid 100051 td 0xc690a000
sched_switch(c690a000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabfca14
mi_switch(104,0,0,0,c690a000,c1fa3ec4) at mi_switch+0x122/frame 0xdabfca4c
sleepq_switch(c690a000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdabfca74
sleepq_wait(c1fa3ec4,74,c148589d,0,0,...) at sleepq_wait+0x3f/frame 0xdabfcaa0
_sleep(c1fa3ec4,c1fa3eac,74,c148589d,0,...) at _sleep+0x2ae/frame 0xdabfcae8
vm_daemon(0,dabfcba8,0,0,0,...) at vm_daemon+0x130/frame 0xdabfcb64
fork_exit(c1072d80,0,dabfcba8) at fork_exit+0xa3/frame 0xdabfcb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabfcb94
--- trap 0, eip = 0, esp = 0xdabfcbe0, ebp = 0 ---

Tracing command pagedaemon pid 7 tid 100057 td 0xc6d97000
sched_switch(c6d97000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xe9acca64
mi_switch(104,0,0,0,c6d97000,c1fa3aa4) at mi_switch+0x122/frame 0xe9acca9c
sleepq_switch(c6d97000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xe9accac4
sleepq_wait(c1fa3aa4,54,c14b4e9f,0,0,...) at sleepq_wait+0x3f/frame 0xe9accaf0
_sleep(c1fa3aa4,c1fa3a90,54,c14b4e9f,0,...) at _sleep+0x2ae/frame 0xe9accb38
uma_reclaim_worker(0,e9accba8,0,0,0,...) at uma_reclaim_worker+0xb0/frame 0xe9accb64
fork_exit(c10545c0,0,e9accba8) at fork_exit+0xa3/frame 0xe9accb94
fork_trampoline() at fork_trampoline+0x8/frame 0xe9accb94

--- trap 0, eip = 0, esp = 0xe9accbe0, ebp = 0 ---

Tracing command pagedaemon pid 7 tid 100050 td 0xc690a320
sched_switch(c690a320,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xdabf9a20
mi_switch(104,0,101,c690a320,dabf9aac,c2027604) at mi_switch+0x122/frame 0xdabf9a58
sleepq_switch(c690a320,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdabf9a80
sleepq_timedwait(c2027604,54,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdabf9aac
_sleep(c2027604,c2027580,54,c148589d,fffffed8,...) at _sleep+0x282/frame 0xdabf9af4
vm_pageout(0,dabf9ba8,0,0,0,...) at vm_pageout+0x23b/frame 0xdabf9b64
fork_exit(c1073b20,0,dabf9ba8) at fork_exit+0xa3/frame 0xdabf9b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabf9b94
--- trap 0, eip = 0, esp = 0xdabf9be0, ebp = 0 ---

Tracing command sctp_iterator pid 6 tid 100046 td 0xc690b000
sched_switch(c690b000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabeda64
mi_switch(104,0,0,0,c690b000,c2021f74) at mi_switch+0x122/frame 0xdabeda9c
sleepq_switch(c690b000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdabedac4
sleepq_wait(c2021f74,0,c149c033,0,0,...) at sleepq_wait+0x3f/frame 0xdabedaf0
_sleep(c2021f74,c2021f3c,0,c149c033,0,...) at _sleep+0x2ae/frame 0xdabedb38
sctp_iterator_thread(0,dabedba8,0,0,0,...) at sctp_iterator_thread+0x9c/frame 0xdabedb64
fork_exit(c0ea87d0,0,dabedba8) at fork_exit+0xa3/frame 0xdabedb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabedb94
--- trap 0, eip = 0, esp = 0xdabedbe0, ebp = 0 ---

Tracing command pf purge pid 5 tid 100044 td 0xc690b640
sched_switch(c690b640,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xdabe6a58
mi_switch(104,0,101,c690b640,dabe6ae4,c0f94ba0) at mi_switch+0x122/frame 0xdabe6a90
sleepq_switch(c690b640,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdabe6ab8
sleepq_timedwait(c0f94ba0,0,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdabe6ae4
_sleep(c0f94ba0,c2024ef4,0,c14a8087,1999997c,...) at _sleep+0x282/frame 0xdabe6b2c
pf_purge_thread(0,dabe6ba8,0,0,0,...) at pf_purge_thread+0xba/frame 0xdabe6b64
fork_exit(c0f94ba0,0,dabe6ba8) at fork_exit+0xa3/frame 0xdabe6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabe6b94
--- trap 0, eip = 0, esp = 0xdabe6be0, ebp = 0 ---

Tracing command usb pid 16 tid 100041 td 0xc690c000
sched_switch(c690c000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabdda78
mi_switch(104,0,0,0,c690c000,c6893d64) at mi_switch+0x122/frame 0xdabddab0

sleepq_switch(c690c000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdabddad8
sleepq_wait(c6893d64,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabddb04
_cv_wait(c6893d64,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabddb34
usb_process(c6893d5c,dabddba8,0,0,0,...) at usb_process+0xf6/frame 0xdabddb64
fork_exit(c0ab1850,c6893d5c,dabddba8) at fork_exit+0xa3/frame 0xdabddb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabddb94
--- trap 0, eip = 0, esp = 0xdabddbe0, ebp = 0 ---

Tracing command usb pid 16 tid 100040 td 0xc690c320
sched_switch(c690c320,0,104,4000c,0,...) at sched_switch+0x2d2/frame 0xdabdaa78
mi_switch(104,0,c68d2fc8,c0d1ba1d,c690c320,c6893d34) at mi_switch+0x122/frame 0xdabdaab0
sleepq_switch(c690c320,0,c1480b9a,269,500,...) at sleepq_switch+0x15b/frame 0xdabdaad8
sleepq_wait(c6893d34,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabdab04
_cv_wait(c6893d34,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabdab34
usb_process(c6893d2c,dabdaba8,0,0,0,...) at usb_process+0xf6/frame 0xdabdab64
fork_exit(c0ab1850,c6893d2c,dabdaba8) at fork_exit+0xa3/frame 0xdabdab94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabdab94
--- trap 0, eip = 0, esp = 0xdabdabe0, ebp = 0 ---

Tracing command usb pid 16 tid 100039 td 0xc6857320
sched_switch(c6857320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabd7a78
mi_switch(104,0,0,0,c6857320,c6893d04) at mi_switch+0x122/frame 0xdabd7ab0
sleepq_switch(c6857320,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdabd7ad8
sleepq_wait(c6893d04,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabd7b04
_cv_wait(c6893d04,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabd7b34
usb_process(c6893cfc,dabd7ba8,0,0,0,...) at usb_process+0xf6/frame 0xdabd7b64
fork_exit(c0ab1850,c6893cfc,dabd7ba8) at fork_exit+0xa3/frame 0xdabd7b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabd7b94
--- trap 0, eip = 0, esp = 0xdabd7be0, ebp = 0 ---

Tracing command usb pid 16 tid 100038 td 0xc6857640
sched_switch(c6857640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabd4a78
mi_switch(104,0,0,0,c6857640,c6893cd4) at mi_switch+0x122/frame 0xdabd4ab0
sleepq_switch(c6857640,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdabd4ad8
sleepq_wait(c6893cd4,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabd4b04
_cv_wait(c6893cd4,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabd4b34
usb_process(c6893ccc,dabd4ba8,0,0,0,...) at usb_process+0xf6/frame 0xdabd4b64
fork_exit(c0ab1850,c6893ccc,dabd4ba8) at fork_exit+0xa3/frame 0xdabd4b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabd4b94
--- trap 0, eip = 0, esp = 0xdabd4be0, ebp = 0 ---

Tracing command usb pid 16 tid 100037 td 0xc6857960

sched_switch(c6857960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabd1a78
mi_switch(104,0,0,0,c6857960,c6893ca4) at mi_switch+0x122/frame 0xdabd1ab0
sleepq_switch(c6857960,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdabd1ad8
sleepq_wait(c6893ca4,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabd1b04
_cv_wait(c6893ca4,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabd1b34
usb_process(c6893c9c,dabd1ba8,0,0,0,...) at usb_process+0xf6/frame 0xdabd1b64
fork_exit(c0ab1850,c6893c9c,dabd1ba8) at fork_exit+0xa3/frame 0xdabd1b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabd1b94
--- trap 0, eip = 0, esp = 0xdabd1be0, ebp = 0 ---

Tracing command usb pid 16 tid 100033 td 0xc68b4640
sched_switch(c68b4640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab83a78
mi_switch(104,0,0,0,c68b4640,c67fcb8c) at mi_switch+0x122/frame 0xdab83ab0
sleepq_switch(c68b4640,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdab83ad8
sleepq_wait(c67fcb8c,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab83b04
_cv_wait(c67fcb8c,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab83b34
usb_process(c67fcb84,dab83ba8,0,0,0,...) at usb_process+0xf6/frame 0xdab83b64
fork_exit(c0ab1850,c67fcb84,dab83ba8) at fork_exit+0xa3/frame 0xdab83b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab83b94
--- trap 0, eip = 0, esp = 0xdab83be0, ebp = 0 ---

Tracing command usb pid 16 tid 100032 td 0xc68b4960
sched_switch(c68b4960,0,104,4000c,0,...) at sched_switch+0x2d2/frame 0xdab80a78
mi_switch(104,0,c68b0fc8,c0d1ba1d,c68b4960,c67fcb5c) at mi_switch+0x122/frame
0xdab80ab0
sleepq_switch(c68b4960,0,c1480b9a,269,100,...) at sleepq_switch+0x15b/frame
0xdab80ad8
sleepq_wait(c67fcb5c,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab80b04
_cv_wait(c67fcb5c,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab80b34
usb_process(c67fcb54,dab80ba8,0,0,0,...) at usb_process+0xf6/frame 0xdab80b64
fork_exit(c0ab1850,c67fcb54,dab80ba8) at fork_exit+0xa3/frame 0xdab80b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab80b94
--- trap 0, eip = 0, esp = 0xdab80be0, ebp = 0 ---

Tracing command usb pid 16 tid 100031 td 0xc68b4c80
sched_switch(c68b4c80,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab7da78
mi_switch(104,0,0,0,c68b4c80,c67fcb2c) at mi_switch+0x122/frame 0xdab7dab0
sleepq_switch(c68b4c80,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdab7dad8
sleepq_wait(c67fcb2c,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab7db04
_cv_wait(c67fcb2c,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab7db34
usb_process(c67fcb24,dab7dba8,0,0,0,...) at usb_process+0xf6/frame 0xdab7db64
fork_exit(c0ab1850,c67fcb24,dab7dba8) at fork_exit+0xa3/frame 0xdab7db94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab7db94
--- trap 0, eip = 0, esp = 0xdab7dbe0, ebp = 0 ---

Tracing command usb pid 16 tid 100030 td 0xc68b9000
sched_switch(c68b9000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab7aa78
mi_switch(104,0,0,0,c68b9000,c67fcafc) at mi_switch+0x122/frame 0xdab7aab0
sleepq_switch(c68b9000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdab7aad8
sleepq_wait(c67fcafc,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab7ab04
_cv_wait(c67fcafc,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab7ab34
usb_process(c67fcaf4,dab7aba8,0,0,0,...) at usb_process+0xf6/frame 0xdab7ab64
fork_exit(c0ab1850,c67fcaf4,dab7aba8) at fork_exit+0xa3/frame 0xdab7ab94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab7ab94
--- trap 0, eip = 0, esp = 0xdab7abe0, ebp = 0 ---

Tracing command usb pid 16 tid 100029 td 0xc68b9320
sched_switch(c68b9320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab77a78
mi_switch(104,0,0,0,c68b9320,c67fcacc) at mi_switch+0x122/frame 0xdab77ab0
sleepq_switch(c68b9320,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdab77ad8
sleepq_wait(c67fcacc,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab77b04
_cv_wait(c67fcacc,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab77b34
usb_process(c67fcac4,dab77ba8,0,0,0,...) at usb_process+0xf6/frame 0xdab77b64
fork_exit(c0ab1850,c67fcac4,dab77ba8) at fork_exit+0xa3/frame 0xdab77b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab77b94
--- trap 0, eip = 0, esp = 0xdab77be0, ebp = 0 ---

Tracing command cam pid 4 tid 100049 td 0xc690a640
sched_switch(c690a640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabf6a3c
mi_switch(104,0,0,0,c690a640,c1eb8eac) at mi_switch+0x122/frame 0xdabf6a74
sleepq_switch(c690a640,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdabf6a9c
sleepq_wait(c1eb8eac,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdabf6ac8
_sleep(c1eb8eac,c1eb8efc,5c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdabf6b10
xpt_scanner_thread(0,dabf6ba8,0,0,0,...) at xpt_scanner_thread+0xcc/frame
0xdabf6b64
fork_exit(c04d0120,0,dabf6ba8) at fork_exit+0xa3/frame 0xdabf6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabf6b94
--- trap 0, eip = 0, esp = 0xdabf6be0, ebp = 0 ---

Tracing command cam pid 4 tid 100017 td 0xc6856640
sched_switch(c6856640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaae0a3c
mi_switch(104,0,0,daae0ad8,c6856640,c1eb9000) at mi_switch+0x122/frame
0xdaae0a74
sleepq_switch(c6856640,0,c1480b9a,269,c6d80000,...) at sleepq_switch+0x15b/frame
0xdaae0a9c
sleepq_wait(c1eb9000,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaae0ac8
_sleep(c1eb9000,c1eb8f80,5c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaae0b10
xpt_done_td(c1eb8f80,daae0ba8,0,0,0,...) at xpt_done_td+0xce/frame 0xdaae0b64

fork_exit(c04cfd00,c1eb8f80,daae0ba8) at fork_exit+0xa3/frame 0xdaae0b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaae0b94
--- trap 0, eip = 0, esp = 0xdaae0be0, ebp = 0 ---

Tracing command rand_harvestq pid 15 tid 100014 td 0xc6857000
sched_switch(c6857000,0,104,0,c201e5a4,...) at sched_switch+0x2d2/frame 0xdaad7a54
mi_switch(104,0,5,c6857000,daad7ae0,c1ed60c8) at mi_switch+0x122/frame 0xdaad7a8c
sleepq_switch(c6857000,0,c1480b9a,28c,5,...) at sleepq_switch+0x15b/frame 0xdaad7ab4
sleepq_timedwait(c1ed60c8,0,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdaad7ae0
msleep_spin_sbt(c1ed60c8,c2015ee0,c1467e98,19999999,0,...) at msleep_spin_sbt+0x1c0/frame 0xdaad7b18
random_kthread(c0989220,daad7ba8,0,0,0,...) at random_kthread+0x2a2/frame 0xdaad7b64
fork_exit(c09883d0,c0989220,daad7ba8) at fork_exit+0xa3/frame 0xdaad7b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaad7b94
--- trap 0, eip = 0, esp = 0xdaad7be0, ebp = 0 ---

Tracing command crypto returns pid 3 tid 100013 td 0xc6710000
sched_switch(c6710000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaad3a44
mi_switch(104,0,0,0,c6710000,c1fa2e28) at mi_switch+0x122/frame 0xdaad3a7c
sleepq_switch(c6710000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdaad3aa4
sleepq_wait(c1fa2e28,6c,c14ac401,0,0,...) at sleepq_wait+0x3f/frame 0xdaad3ad0
_sleep(c1fa2e28,c1fa2e14,6c,c14ac401,0,...) at _sleep+0x2ae/frame 0xdaad3b18
crypto_ret_proc(0,daad3ba8,0,0,0,...) at crypto_ret_proc+0x20a/frame 0xdaad3b64
fork_exit(c0fd4c30,0,daad3ba8) at fork_exit+0xa3/frame 0xdaad3b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaad3b94
--- trap 0, eip = 0, esp = 0xdaad3be0, ebp = 0 ---

Tracing command crypto pid 2 tid 100012 td 0xc6710320
sched_switch(c6710320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaad0a50
mi_switch(104,0,0,0,c6710320,c1fa2d68) at mi_switch+0x122/frame 0xdaad0a88
sleepq_switch(c6710320,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdaad0ab0
sleepq_wait(c1fa2d68,6c,c14ac411,0,0,...) at sleepq_wait+0x3f/frame 0xdaad0adc
_sleep(c1fa2d68,c1fa2d50,6c,c14ac411,0,...) at _sleep+0x2ae/frame 0xdaad0b24
crypto_proc(0,daad0ba8,0,0,0,...) at crypto_proc+0x2e8/frame 0xdaad0b64
fork_exit(c0fd48e0,0,daad0ba8) at fork_exit+0xa3/frame 0xdaad0b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaad0b94
--- trap 0, eip = 0, esp = 0xdaad0be0, ebp = 0 ---

Tracing command geom pid 14 tid 100010 td 0xc6710960
sched_switch(c6710960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaacaa40
mi_switch(104,0,c6710960,daacaa98,c6710960,c201cfd8) at mi_switch+0x122/frame

0xdaacaa78
sleepq_switch(c6710960,0,c1480b9a,269,c6710960,...) at sleepq_switch+0x15b/frame 0xdaacaaa0
sleepq_wait(c201cfd8,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaacaacc
_sleep(c201cfd8,c1f76e88,25c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaacab14
g_io_schedule_down(c6710960,5c,c146ae6d,6b,daacab94,...) at g_io_schedule_down+0x5c/frame 0xdaacab4c
g_down_procbody(0,daacaba8,0,0,0,...) at g_down_procbody+0x6d/frame 0xdaacab64
fork_exit(c0c56dd0,0,daacaba8) at fork_exit+0xa3/frame 0xdaacab94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaacab94
--- trap 0, eip = 0, esp = 0xdaacabe0, ebp = 0 ---

Tracing command geom pid 14 tid 100009 td 0xc6710c80
sched_switch(c6710c80,0,104,c65b7e40,0,...) at sched_switch+0x2d2/frame 0xdaac7a40
mi_switch(104,0,8,c6710c80,c6710c80,c201cfd4) at mi_switch+0x122/frame 0xdaac7a78
sleepq_switch(c6710c80,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdaac7aa0
sleepq_wait(c201cfd4,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaac7acc
_sleep(c201cfd4,c1f76ea8,25c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaac7b14
g_io_schedule_up(c6710c80,5c,c146ae6d,5e,daac7b94,...) at g_io_schedule_up+0xd5/frame 0xdaac7b4c
g_up_procbody(0,daac7ba8,0,0,0,...) at g_up_procbody+0x6d/frame 0xdaac7b64
fork_exit(c0c56d60,0,daac7ba8) at fork_exit+0xa3/frame 0xdaac7b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaac7b94
--- trap 0, eip = 0, esp = 0xdaac7be0, ebp = 0 ---

Tracing command geom pid 14 tid 100008 td 0xc6711000
sched_switch(c6711000,0,104,0,c6d96c50,...) at sched_switch+0x2d2/frame 0xdaac4a54
mi_switch(104,0,c0d1bc92,c6d96960,c6711000,c201cfd0) at mi_switch+0x122/frame 0xdaac4a8c
sleepq_switch(c6711000,0,c1480b9a,269,daac4ad8,...) at sleepq_switch+0x15b/frame 0xdaac4ab4
sleepq_wait(c201cfd0,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaac4ae0
_sleep(c201cfd0,c1f76e60,25c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaac4b28
g_run_events(0,daac4ba8,0,0,0,...) at g_run_events+0x62/frame 0xdaac4b64
fork_exit(c0c56cf0,0,daac4ba8) at fork_exit+0xa3/frame 0xdaac4b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaac4b94
--- trap 0, eip = 0, esp = 0xdaac4be0, ebp = 0 ---

Tracing command ng_queue pid 13 tid 100007 td 0xc6711320
sched_switch(c6711320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaac1a50
mi_switch(104,0,0,0,c6711320,c1e9081c) at mi_switch+0x122/frame 0xdaac1a88
sleepq_switch(c6711320,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdaac1ab0

sleepq_wait(c1e9081c,8,c1485c29,0,0,...) at sleepq_wait+0x3f/frame 0xdaac1adc
_sleep(c1e9081c,c1f9cac0,8,c1485c29,0,...) at _sleep+0x2ae/frame 0xdaac1b24
ngthread(0,daac1ba8,0,0,0,...) at ngthread+0xbc/frame 0xdaac1b64
fork_exit(c0e55280,0,daac1ba8) at fork_exit+0xa3/frame 0xdaac1b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaac1b94
--- trap 0, eip = 0, esp = 0xdaac1be0, ebp = 0 ---

Tracing command intr pid 12 tid 100047 td 0xc690ac80
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100045 td 0xc690b320
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100043 td 0xc690b960
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100042 td 0xc690bc80
sched_switch(c690bc80,0,109,c1fa5808,c690bc80,...) at sched_switch+0x2d2/frame
0xdabe0af0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdabe0b28
ithread_loop(c68eced0,dabe0ba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdabe0b64
fork_exit(c0cd81d0,c68eced0,dabe0ba8) at fork_exit+0xa3/frame 0xdabe0b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabe0b94
--- trap 0, eip = 0, esp = 0xdabe0be0, ebp = 0 ---

Tracing command intr pid 12 tid 100036 td 0xc6857c80
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100028 td 0xc68b9640
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100026 td 0xc6716960
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100025 td 0xc6716c80
sched_switch(c6716c80,0,109,c6854800,0,...) at sched_switch+0x2d2/frame
0xdab20af0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdab20b28
ithread_loop(c688a4a0,dab20ba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdab20b64
fork_exit(c0cd81d0,c688a4a0,dab20ba8) at fork_exit+0xa3/frame 0xdab20b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab20b94
--- trap 0, eip = 0, esp = 0xdab20be0, ebp = 0 ---

Tracing command intr pid 12 tid 100024 td 0xc6855000
sched_switch(c6855000,0,109,0,0,...) at sched_switch+0x2d2/frame 0xdaaf5af0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdaaf5b28
ithread_loop(c65bbd60,daaf5ba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdaaf5b64

fork_exit(c0cd81d0,c65bbd60,daaf5ba8) at fork_exit+0xa3/frame 0xdaaf5b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaf5b94
--- trap 0, eip = 0, esp = 0xdaaf5be0, ebp = 0 ---

Tracing command intr pid 12 tid 100022 td 0xc6855640
sched_switch(c6855640,0,109,0,0,...) at sched_switch+0x2d2/frame 0xdaaefaf0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdaaefb28
ithread_loop(c65bbd80,daaefba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdaaefb64
fork_exit(c0cd81d0,c65bbd80,daaefba8) at fork_exit+0xa3/frame 0xdaaefb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaefb94
--- trap 0, eip = 0, esp = 0xdaaefbe0, ebp = 0 ---

Tracing command intr pid 12 tid 100016 td 0xc6856960
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100006 td 0xc6711640
sched_switch(c6711640,0,109,c201e500,0,...) at sched_switch+0x2d2/frame
0xdaabeaf0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdaabeb28
ithread_loop(c65ba0c0,daabeba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdaabeb64
fork_exit(c0cd81d0,c65ba0c0,daabeba8) at fork_exit+0xa3/frame 0xdaabeb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaabeb94
--- trap 0, eip = 0, esp = 0xdaabebe0, ebp = 0 ---

Tracing command intr pid 12 tid 100005 td 0xc6711960
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100004 td 0xc6711c80
fork_trampoline() at fork_trampoline

Tracing command idle pid 11 tid 100003 td 0xc6716000
sched_switch(c6716000,0,608,0,56f000,...) at sched_switch+0x2d2/frame 0xdaab5a94
mi_switch(608,0,c147cb2c,d3,c1f86f04,...) at mi_switch+0x122/frame 0xdaab5acc
critical_exit(0,0,0,0,0,...) at critical_exit+0x89/frame 0xdaab5ae8
sched_idletd(0,daab5ba8,0,0,0,...) at sched_idletd+0x1dd/frame 0xdaab5b64
fork_exit(c0d3f8b0,0,daab5ba8) at fork_exit+0xa3/frame 0xdaab5b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaab5b94
--- trap 0, eip = 0, esp = 0xdaab5be0, ebp = 0 ---

Tracing command init pid 1 tid 100002 td 0xc6716320
sched_switch(c6716320,0,104,c105928c,0,...) at sched_switch+0x2d2/frame
0xdaab1820
mi_switch(104,0,daab18d0,c0cf40f2,c1f86e80,c6716320) at mi_switch+0x122/frame
0xdaab1858
sleepq_switch(c6716320,0,c1480b9a,1a8,0,...) at sleepq_switch+0x15b/frame
0xdaab1880
sleepq_catch_signals(0,c1480b9a,154,0,0,...) at sleepq_catch_signals+0x5be/frame

0xdaab18d4
sleepq_wait_sig(c670f308,6c,c1481bd3,100,0,...) at sleepq_wait_sig+0x14/frame 0xdaab1900
_sleep(c670f308,c670f3a0,16c,c1481bd3,0,...) at _sleep+0x29b/frame 0xdaab1948
kern_wait6(c6716320,7,0,0,daab19e4,...) at kern_wait6+0x71a/frame 0xdaab19b8
sys_wait4(c6716320,daab1b68,c670f308,c6716320,c1e73e50,...) at sys_wait4+0x94/frame 0xdaab1ad8
syscall(daab1ba8) at syscall+0x5c9/frame 0xdaab1b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xdaab1b9c
--- syscall (7, FreeBSD ELF32, sys_wait4), eip = 0x8072c67, esp = 0xbfbfe8bc, ebp = 0xbfbfe8d0 ---

Tracing command audit pid 10 tid 100001 td 0xc6716640
sched_switch(c6716640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaaaea54
mi_switch(104,0,0,0,c6716640,c2025edc) at mi_switch+0x122/frame 0xdaaaea8c
sleepq_switch(c6716640,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdaaaeab4
sleepq_wait(c2025edc,0,c14ad51d,1,0,...) at sleepq_wait+0x3f/frame 0xdaaaeae0
_cv_wait(c2025edc,c2025e68,0,0,0,...) at _cv_wait+0x192/frame 0xdaaaeb10
audit_worker(0,daaaeba8,0,0,0,...) at audit_worker+0xa4/frame 0xdaaaeb64
fork_exit(c0ff7340,0,daaaeba8) at fork_exit+0xa3/frame 0xdaaaeb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaaeb94
--- trap 0, eip = 0, esp = 0xdaaaebe0, ebp = 0 ---

Tracing command kernel pid 0 tid 100048 td 0xc690a960
sched_switch(c690a960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabf3a58
mi_switch(104,0,dabf3ac8,c0d5f211,c690a960,c6712980) at mi_switch+0x122/frame 0xdabf3a90
sleepq_switch(c690a960,0,c1480b9a,269,c6712990,...) at sleepq_switch+0x15b/frame 0xdabf3ab8
sleepq_wait(c6712980,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdabf3ae4
_sleep(c6712980,c6712998,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdabf3b2c
taskqueue_thread_loop(c1eb8f24,dabf3ba8,0,0,0,...) at taskqueue_thread_loop+0x11b/frame 0xdabf3b64
fork_exit(c0d63900,c1eb8f24,dabf3ba8) at fork_exit+0xa3/frame 0xdabf3b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabf3b94
--- trap 0, eip = 0, esp = 0xdabf3be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100035 td 0xc68b4000
sched_switch(c68b4000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabc9a68
mi_switch(104,0,0,0,c68b4000,c68d0480) at mi_switch+0x122/frame 0xdabc9aa0
sleepq_switch(c68b4000,0,c1480b9a,269,c68d0480,...) at sleepq_switch+0x15b/frame 0xdabc9ac8
sleepq_wait(c68d0480,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdabc9af4
msleep_spin_sbt(c68d0480,c68d0498,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/frame 0xdabc9b2c
taskqueue_thread_loop(c68e35dc,dabc9ba8,0,0,0,...) at

taskqueue_thread_loop+0x15c/frame 0xdabc9b64
fork_exit(c0d63900,c68e35dc,dabc9ba8) at fork_exit+0xa3/frame 0xdabc9b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabc9b94
--- trap 0, eip = 0, esp = 0xdabc9be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100034 td 0xc68b4320
sched_switch(c68b4320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaba6a68
mi_switch(104,0,0,0,c68b4320,c68d0700) at mi_switch+0x122/frame 0xdaba6aa0
sleepq_switch(c68b4320,0,c1480b9a,269,c68d0700,...) at sleepq_switch+0x15b/frame
0xdaba6ac8
sleepq_wait(c68d0700,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaba6af4
msleep_spin_sbt(c68d0700,c68d0718,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/
frame 0xdaba6b2c
taskqueue_thread_loop(c68cc5dc,daba6ba8,0,0,0,...) at
taskqueue_thread_loop+0x15c/frame 0xdaba6b64
fork_exit(c0d63900,c68cc5dc,daba6ba8) at fork_exit+0xa3/frame 0xdaba6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaba6b94
--- trap 0, eip = 0, esp = 0xdaba6be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100027 td 0xc68b9960
sched_switch(c68b9960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab68a68
mi_switch(104,0,0,0,c68b9960,c6883300) at mi_switch+0x122/frame 0xdab68aa0
sleepq_switch(c68b9960,0,c1480b9a,269,c6883300,...) at sleepq_switch+0x15b/frame
0xdab68ac8
sleepq_wait(c6883300,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdab68af4
msleep_spin_sbt(c6883300,c6883318,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/
frame 0xdab68b2c
taskqueue_thread_loop(c68885dc,dab68ba8,0,0,0,...) at
taskqueue_thread_loop+0x15c/frame 0xdab68b64
fork_exit(c0d63900,c68885dc,dab68ba8) at fork_exit+0xa3/frame 0xdab68b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab68b94
--- trap 0, eip = 0, esp = 0xdab68be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100023 td 0xc6855320
sched_switch(c6855320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaaf2a58
mi_switch(104,0,daaf2ac8,c0d5f211,c6855320,c6712400) at mi_switch+0x122/frame
0xdaaf2a90
sleepq_switch(c6855320,0,c1480b9a,269,c6712410,...) at sleepq_switch+0x15b/frame
0xdaaf2ab8
sleepq_wait(c6712400,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaaf2ae4
_sleep(c6712400,c6712418,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaaf2b2c
taskqueue_thread_loop(c2025ee8,daaf2ba8,0,0,0,...) at taskqueue_thread_loop+0x11b/
frame 0xdaaf2b64
fork_exit(c0d63900,c2025ee8,daaf2ba8) at fork_exit+0xa3/frame 0xdaaf2b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaf2b94
--- trap 0, eip = 0, esp = 0xdaaf2be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100021 td 0xc6855960
sched_switch(c6855960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaaeca58
mi_switch(104,0,daaecac8,c0d5f211,c6855960,c6712680) at mi_switch+0x122/frame 0xdaaeca90
sleepq_switch(c6855960,0,c1480b9a,269,c6712690,...) at sleepq_switch+0x15b/frame 0xdaaecab8
sleepq_wait(c6712680,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaaecae4
_sleep(c6712680,c6712698,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaaecb2c
taskqueue_thread_loop(c201d81c,daaecba8,0,0,0,...) at taskqueue_thread_loop+0x11b/frame 0xdaaecb64
fork_exit(c0d63900,c201d81c,daaecba8) at fork_exit+0xa3/frame 0xdaaecb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaecb94
--- trap 0, eip = 0, esp = 0xdaaecbe0, ebp = 0 ---

Tracing command kernel pid 0 tid 100020 td 0xc6855c80
sched_switch(c6855c80,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaae9a68
mi_switch(104,0,0,0,c6855c80,c6712700) at mi_switch+0x122/frame 0xdaae9aa0
sleepq_switch(c6855c80,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdaae9ac8
sleepq_wait(c6712700,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaae9af4
msleep_spin_sbt(c6712700,c6712718,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/frame 0xdaae9b2c
taskqueue_thread_loop(c1ff1a74,daae9ba8,0,0,0,...) at taskqueue_thread_loop+0x15c/frame 0xdaae9b64
fork_exit(c0d63900,c1ff1a74,daae9ba8) at fork_exit+0xa3/frame 0xdaae9b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaae9b94
--- trap 0, eip = 0, esp = 0xdaae9be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100019 td 0xc6856000
sched_switch(c6856000,0,104,6681622a,0,...) at sched_switch+0x2d2/frame 0xdaae6a68
mi_switch(104,0,c6884c80,0,c6856000,c6712700) at mi_switch+0x122/frame 0xdaae6aa0
sleepq_switch(c6856000,0,c1480b9a,269,c6712700,...) at sleepq_switch+0x15b/frame 0xdaae6ac8
sleepq_wait(c6712700,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaae6af4
msleep_spin_sbt(c6712700,c6712718,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/frame 0xdaae6b2c
taskqueue_thread_loop(c1ff1a74,daae6ba8,0,0,0,...) at taskqueue_thread_loop+0x15c/frame 0xdaae6b64
fork_exit(c0d63900,c1ff1a74,daae6ba8) at fork_exit+0xa3/frame 0xdaae6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaae6b94
--- trap 0, eip = 0, esp = 0xdaae6be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100018 td 0xc6856320
sched_switch(c6856320,0,104,20,0,...) at sched_switch+0x2d2/frame 0xdaae3a68
mi_switch(104,0,0,0,c6856320,c6712700) at mi_switch+0x122/frame 0xdaae3aa0

sleepq_switch(c6856320,0,c1480b9a,269,c6712700,...) at sleepq_switch+0x15b/frame 0xdaae3ac8
sleepq_wait(c6712700,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaae3af4
msleep_spin_sbt(c6712700,c6712718,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/frame 0xdaae3b2c
taskqueue_thread_loop(c1ff1a74,daae3ba8,0,0,0,...) at taskqueue_thread_loop+0x15c/frame 0xdaae3b64
fork_exit(c0d63900,c1ff1a74,daae3ba8) at fork_exit+0xa3/frame 0xdaae3b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaae3b94
--- trap 0, eip = 0, esp = 0xdaae3be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100015 td 0xc6856c80
sched_switch(c6856c80,0,104,c0cb43f3,0,...) at sched_switch+0x2d2/frame 0xdaadaa58
mi_switch(104,0,daadaac8,c0d5f211,c6856c80,c6712b00) at mi_switch+0x122/frame 0xdaadaa90
sleepq_switch(c6856c80,0,c1480b9a,269,c6712b10,...) at sleepq_switch+0x15b/frame 0xdaadaab8
sleepq_wait(c6712b00,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaadaae4
_sleep(c6712b00,c6712b18,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaadab2c
taskqueue_thread_loop(c2021718,daadaba8,0,0,0,...) at taskqueue_thread_loop+0x11b/frame 0xdaadab64
fork_exit(c0d63900,c2021718,daadaba8) at fork_exit+0xa3/frame 0xdaadab94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaadab94
--- trap 0, eip = 0, esp = 0xdaadabe0, ebp = 0 ---

Tracing command kernel pid 0 tid 100011 td 0xc6710640
sched_switch(c6710640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaacda58
mi_switch(104,0,daacdac8,c0d5f211,c6710640,c6713e80) at mi_switch+0x122/frame 0xdaacda90
sleepq_switch(c6710640,0,c1480b9a,269,c6713e90,...) at sleepq_switch+0x15b/frame 0xdaacdab8
sleepq_wait(c6713e80,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaacdae4
_sleep(c6713e80,c6713e98,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaacdb2c
taskqueue_thread_loop(c1f94fc4,daacdba8,0,0,0,...) at taskqueue_thread_loop+0x11b/frame 0xdaacdb64
fork_exit(c0d63900,c1f94fc4,daacdba8) at fork_exit+0xa3/frame 0xdaacdb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaacdb94
--- trap 0, eip = 0, esp = 0xdaacdbe0, ebp = 0 ---

Tracing command kernel pid 0 tid 100000 td 0xc201d360
sched_switch(c201d360,0,104,0,c201e5a4,...) at sched_switch+0x2d2/frame 0xc2422adc
mi_switch(104,0,101,c201d360,c2422b68,c201d054) at mi_switch+0x122/frame 0xc2422b14
sleepq_switch(c201d360,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xc2422b3c

```
sleepq_timedwait(c201d054,54,9,0,0,...) at sleepq_timedwait+0x3f/frame 0xc2422b68
_sleep(c201d054,0,54,c14b5ec1,fffff470,...) at _sleep+0x282/frame 0xc2422bb0
swapper() at swapper+0x2c0/frame 0xc2422bf8
begin() at begin+0x22
db:0:kdb.enter.default>  capture off
config.txt0600001367012754764136  7561 ustarrootwheeloptions
CONFIG_AUTOGENERATED
ident   pfSense
machine     i386
cpu     I686_CPU
cpu     I586_CPU
cpu     I486_CPU
makeoptions WITH_CTF=1
makeoptions DEBUG=-g
options     CONSPEED=115200
options     MROUTING
options     DEVICE_POLLING
options     HIFN_RNDTEST
options     HIFN_DEBUG
options     MSGTQL=2048
options     MSGSSZ=32
options     MSGSEG=512
options     MSGMNI=40
options     MSGMNB=8192
options     ALTQ_CODEL
options     ALTQ_NOPCC
options     ALTQ_FAIRQ
options     ALTQ_PRIQ
options     ALTQ_HFSC
options     ALTQ_RIO
options     ALTQ_RED
options     ALTQ_CBQ
options     ALTQ
options     TCP_SIGNATURE
options     IPSEC_NAT_T
options     IPSEC
options     NETGRAPH_PRED1
options     NETGRAPH_DEFLATE
options     NETGRAPH_CAR
options     NETGRAPH_PIPE
options     NETGRAPH_TCPMSS
options     NETGRAPH_TEE
options     NETGRAPH_HOLE
options     NETGRAPH_FRAME_RELAY
options     NETGRAPH_ASYNC
options     NETGRAPH_ECHO
options     NETGRAPH_CISCO
```

```
options         NETGRAPH_BRIDGE
options         NETGRAPH_ONE2MANY
options         NETGRAPH_LMI
options         NETGRAPH_KSOCKET
options         NETGRAPH_VJC
options         NETGRAPH_UI
options         NETGRAPH_MPPC_ENCRYPTION
options         NETGRAPH_TTY
options         NETGRAPH_SOCKET
options         NETGRAPH_RFC1490
options         NETGRAPH_PPTPGRE
options         NETGRAPH_PPPOE
options         NETGRAPH_PPP
options         NETGRAPH_EIFACE
options         NETGRAPH_IFACE
options         NETGRAPH_ETHER
options         NETGRAPH_BPF
options         NETGRAPH_L2TP
options         NETGRAPH_VLAN
options         NETGRAPH
options         IPSTEALTH
options         IPFIREWALL_VERBOSE
options         IPFIREWALL_DEFAULT_TO_ACCEPT
options         PPS_SYNC
options         NULLFS
options         UNIONFS
options         TMPFS
options         GEOM_BDE
options         GEOM_ELI
options         GEOM_UZIP
options         GEOM_MIRROR
options         DDB
options         XENHVM
options         HYPERV
options         ATH_ENABLE_11N
options         AH_AR5416_INTERRUPT_MITIGATION
options         AH_SUPPORT_AR5416
options         IEEE80211_SUPPORT_MESH
options         IEEE80211_AMPDU_AGE
options         SC_PIXEL_MODE
options         VESA
options         AHD_REG_PRETTY_PRINT
options         AHC_REG_PRETTY_PRINT
options         ATA_STATIC_ID
options         SMP
options         KDB
options         RCTL
```

```
options         RACCT_DEFAULT_TO_DISABLED
options         RACCT
options         INCLUDE_CONFIG_FILE
options         DDB_CTF
options         KDTRACE_HOOKS
options         PROCDESC
options         CAPABILITIES
options         CAPABILITY_MODE
options         AUDIT
options         HWPMC_HOOKS
options         KBD_INSTALL_CDEV
options         PRINTF_BUFR_SIZE=128
options         _KPOSIX_PRIORITY_SCHEDULING
options         SYSVSEM
options         SYSVMSG
options         SYSVSHM
options         STACK
options         KTRACE
options         SCSI_DELAY=5000
options         GEOM_LABEL
options         GEOM_RAID
options         GEOM_PART_GPT
options         PSEUDOFS
options         PROCFS
options         CD9660
options         MSDOSFS
options         NFS_ROOT
options         NFSLOCKD
options         NFSD
options         NFSCL
options         MD_ROOT
options         QUOTA
options         UFS_GJOURNAL
options         UFS_DIRHASH
options         UFS_ACL
options         SOFTUPDATES
options         FFS
options         SCTP
options         TCP_OFFLOAD
options         INET6
options         INET
options         PREEMPTION
options         SCHED_ULE
options         NEW_PCIB
options         NATIVE
options         GEOM_PART_MBR
options         GEOM_PART_EBR_COMPAT
```

```
options         GEOM_PART_EBR
options         GEOM_PART_BSD
options         ISAPNP
device isa
device npx
device mem
device io
device uart_ns8250
device atpic
device apic
device cpufreq
device acpi
device eisa
device pci
device fdc
device ahci
device ata
device mvs
device siis
device ahb
device ahc
device ahd
device esp
device hptiop
device isp
device mpt
device mps
device mpr
device sym
device trm
device adv
device adw
device aha
device aic
device bt
device ncv
device nsp
device stg
device isci
device scbus
device ch
device da
device sa
device cd
device pass
device ses
device amr
```

device arcmsr
device asr
device ciss
device dpt
device hptmv
device hptnr
device hptrr
device hpt27xx
device iir
device ips
device mly
device twa
device tws
device aac
device aacp
device aacraid
device ida
device mfi
device mlx
device mrsas
device pst
device twe
device nvme
device nvd
device atkbdc
device atkbd
device psm
device kbdmux
device vga
device splash
device sc
device vt
device vt_vga
device agp
device pmtimer
device cbb
device pccard
device cardbus
device uart
device ppc
device ppbus
device lpt
device ppi
device puc
device bxe
device de
device em

device igb
device ixgb
device le
device ti
device txp
device vx
device miibus
device ae
device age
device alc
device ale
device bce
device bfe
device bge
device cas
device dc
device et
device fxp
device gem
device hme
device jme
device lge
device msk
device nfe
device nge
device pcn
device re
device rl
device sf
device sge
device sis
device sk
device ste
device stge
device tl
device tx
device vge
device vr
device vte
device wb
device xl
device cs
device ed
device ex
device ep
device fe
device ie

device sn
device xe
device wlan
device wlan_wep
device wlan_ccmp
device wlan_tkip
device wlan_amrr
device an
device ath
device ath_pci
device ath_hal
device ath_rate_sample
device ipw
device iwi
device iwn
device malo
device mwl
device ral
device wi
device wpi
device loop
device random
device padlock_rng
device rdrand_rng
device ether
device vlan
device tun
device md
device gif
device faith
device firmware
device bpf
device uhci
device ohci
device ehci
device xhci
device usb
device ukbd
device umass
device sound
device snd_cmi
device snd_csa
device snd_emu10kx
device snd_es137x
device snd_hda
device snd_ich
device snd_via8233

device mmc
device mmcsd
device sdhci
device virtio
device virtio_pci
device vtnet
device virtio_blk
device virtio_scsi
device virtio_balloon
device hyperv
device xenpci
device vmx
device wlan_rssadapt
device wlan_xauth
device wlan_acl
device iwifw
device ipwfw
device wpifw
device iwnfw
device uath
device ralfw
device ural
device urtw
device rum
device mwlfw
device zyd
device upgt
device udav
device axe
device axge
device aue
device cue
device kue
device mos
device rsu
device rsufw
device run
device runfw
device rue
device urtwn
device urtwnfw
device siba_bwn
device bwn
device bwi
device pty
device ufoma
device ucom

device uslcom
device uplcom
device umct
device uvisor
device uark
device uftdi
device uvscom
device umodem
device u3g
device cdce
device uhid
device firewire
device sbp
device tap
device gre
device if_bridge
device lagg
device netmap
device enc
device pf
device pflog
device carp
device pfsync
device crypto
device cryptodev
device rndtest
device hifn
device ubsec
device safe
device padlock
device speaker
device mxge
device cxgb
device cxgbe
device nve
device oce
device gpioapu
msgbuf.txt0600001747712754764136 7610 ustarrootwheelCopyright (c) 1992-2016 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
	The Regents of the University of California. All rights reserved.
FreeBSD is a registered trademark of The FreeBSD Foundation.
FreeBSD 10.3-RELEASE-p5 #0 7307492(RELENG_2_3_2): Tue Jul 19 13:53:47 CDT 2016
    root@ce23-i386-builder:/builder/pfsense-232/tmp/obj/builder/pfsense-232/tmp/FreeBSD-src/sys/pfSense i386
FreeBSD clang version 3.4.1 (tags/RELEASE_34/dot1-final 208032) 20140512

```
CPU: Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (2893.52-MHz 686-class CPU)
  Origin="GenuineIntel"  Id=0x306a9  Family=0x6  Model=0x3a  Stepping=9

Features=0x783fbbf<FPU,VME,DE,PSE,TSC,MSR,MCE,CX8,APIC,SEP,MTRR,PGE,M
CA,CMOV,PAT,PSE36,MMX,FXSR,SSE,SSE2>

Features2=0x5e98220b<SSE3,PCLMULQDQ,MON,SSSE3,CX16,SSE4.1,SSE4.2,POP
CNT,AESNI,XSAVE,OSXSAVE,AVX,RDRAND>
  AMD Features=0x8000000<RDTSCP>
  AMD Features2=0x1<LAHF>
  TSC: P-state invariant
real memory  = 1073676288 (1023 MB)
avail memory = 1016778752 (969 MB)
pnpbios: Bad PnP BIOS data checksum
random: <Software, Yarrow> initialized
wlan: mac acl policy registered
ipw_bss: You need to read the LICENSE file in /usr/share/doc/legal/intel_ipw/.
ipw_bss: If you agree with the license, set legal.intel_ipw.license_ack=1 in /boot/
loader.conf.
module_register_init: MOD_LOAD (ipw_bss_fw, 0xc081eba0, 0) error 1
ipw_ibss: You need to read the LICENSE file in /usr/share/doc/legal/intel_ipw/.
ipw_ibss: If you agree with the license, set legal.intel_ipw.license_ack=1 in /boot/
loader.conf.
module_register_init: MOD_LOAD (ipw_ibss_fw, 0xc081ec50, 0) error 1
ipw_monitor: You need to read the LICENSE file in /usr/share/doc/legal/intel_ipw/.
ipw_monitor: If you agree with the license, set legal.intel_ipw.license_ack=1 in /boot/
loader.conf.
module_register_init: MOD_LOAD (ipw_monitor_fw, 0xc081ed00, 0) error 1
iwi_bss: You need to read the LICENSE file in /usr/share/doc/legal/intel_iwi/.
iwi_bss: If you agree with the license, set legal.intel_iwi.license_ack=1 in /boot/
loader.conf.
module_register_init: MOD_LOAD (iwi_bss_fw, 0xc08466f0, 0) error 1
iwi_ibss: You need to read the LICENSE file in /usr/share/doc/legal/intel_iwi/.
iwi_ibss: If you agree with the license, set legal.intel_iwi.license_ack=1 in /boot/
loader.conf.
module_register_init: MOD_LOAD (iwi_ibss_fw, 0xc08467a0, 0) error 1
iwi_monitor: You need to read the LICENSE file in /usr/share/doc/legal/intel_iwi/.
iwi_monitor: If you agree with the license, set legal.intel_iwi.license_ack=1 in /boot/
loader.conf.
module_register_init: MOD_LOAD (iwi_monitor_fw, 0xc0846850, 0) error 1
netmap: loaded module
kbd1 at kbdmux0
cryptosoft0: <software crypto> on motherboard
padlock0: No ACE support.
acpi0: <VBOX VBOXXSDT> on motherboard
acpi0: Power Button (fixed)
acpi0: Sleep Button (fixed)
```

attimer0: <AT timer> port 0x40-0x43,0x50-0x53 on acpi0
Timecounter "i8254" frequency 1193182 Hz quality 0
Event timer "i8254" frequency 1193182 Hz quality 100
Timecounter "ACPI-fast" frequency 3579545 Hz quality 900
acpi_timer0: <32-bit timer at 3.579545MHz> port 0x4008-0x400b on acpi0
pcib0: <ACPI Host-PCI bridge> port 0xcf8-0xcff on acpi0
pci_link2: BIOS IRQ 9 for 0.7.INTA does not match previous BIOS IRQ 10
pci0: <ACPI PCI bus> on pcib0
isab0: <PCI-ISA bridge> at device 1.0 on pci0
isa0: <ISA bus> on isab0
atapci0: <Intel PIIX4 UDMA33 controller> port
0x1f0-0x1f7,0x3f6,0x170-0x177,0x376,0xd000-0xd00f at device 1.1 on pci0
ata0: <ATA channel> at channel 0 on atapci0
ata1: <ATA channel> at channel 1 on atapci0
vgapci0: <VGA-compatible display> mem 0xe0000000-0xe07fffff irq 11 at device 2.0 on
pci0
vgapci0: Boot video device
em0: <Intel(R) PRO/1000 Legacy Network Connection 1.1.0> port 0xd010-0xd017 mem
0xf0000000-0xf001ffff irq 10 at device 3.0 on pci0
em0: Ethernet address: 08:00:27:1e:49:46
em0: netmap queues/slots: TX 1/256, RX 1/256
pcm0: <Intel ICH (82801AA)> port 0xd100-0xd1ff,0xd200-0xd23f irq 11 at device 5.0 on
pci0
pcm0: <SigmaTel STAC9700/83/84 AC97 Codec>
ohci0: <OHCI (generic) USB controller> mem 0xf0804000-0xf0804fff irq 11 at device 6.0
on pci0
usbus0 on ohci0
pci0: <bridge> at device 7.0 (no driver attached)
em1: <Intel(R) PRO/1000 Legacy Network Connection 1.1.0> port 0xd240-0xd247 mem
0xf0820000-0xf083ffff irq 9 at device 8.0 on pci0
em1: Ethernet address: 08:00:27:3f:1e:02
em1: netmap queues/slots: TX 1/256, RX 1/256
em2: <Intel(R) PRO/1000 Legacy Network Connection 1.1.0> port 0xd248-0xd24f mem
0xf0840000-0xf085ffff irq 11 at device 9.0 on pci0
em2: Ethernet address: 08:00:27:53:bd:6d
em2: netmap queues/slots: TX 1/256, RX 1/256
ehci0: <Intel 82801FB (ICH6) USB 2.0 controller> mem 0xf0860000-0xf0860fff irq 10 at
device 11.0 on pci0
usbus1: EHCI version 1.0
usbus1 on ehci0
battery0: <ACPI Control Method Battery> on acpi0
acpi_acad0: <AC Adapter> on acpi0
atkbdc0: <Keyboard controller (i8042)> port 0x60,0x64 irq 1 on acpi0
atkbd0: <AT Keyboard> irq 1 on atkbdc0
kbd0 at atkbd0
atkbd0: [GIANT-LOCKED]
psm0: <PS/2 Mouse> irq 12 on atkbdc0

psm0: [GIANT-LOCKED]
psm0: model IntelliMouse Explorer, device ID 4
pmtimer0 on isa0
orm0: <ISA Option ROMs> at iomem 0xc0000-0xc7fff,0xe2000-0xe2fff pnpid ORM0000 on isa0
sc0: <System console> at flags 0x100 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
atrtc0: <AT realtime clock> at port 0x70 irq 8 on isa0
Event timer "RTC" frequency 32768 Hz quality 0
ppc0: parallel port not found.
Timecounters tick every 1.000 msec
pcm0: measured ac97 link rate at 1928 Hz
<5>em0: link state changed to UP
<5>em1: link state changed to UP
<5>em2: link state changed to UP
usbus0: 12Mbps Full Speed USB v1.0
usbus1: 480Mbps High Speed USB v2.0
ugen0.1: <Apple> at usbus0
uhub0: <Apple OHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus0
ugen1.1: <Intel> at usbus1
uhub1: <Intel EHCI root HUB, class 9/0, rev 2.00/1.00, addr 1> on usbus1
ada0 at ata0 bus 0 scbus0 target 0 lun 0
ada0: <VBOX HARDDISK 1.0> ATA-6 device
ada0: Serial Number VB9ddb6727-0736826b
ada0: 33.300MB/s transfers (UDMA2, PIO 65536bytes)
ada0: 5252MB (10757152 512 byte sectors)
ada0: Previously was known as ad0
battery0: critically low charge!
random: unblocking device.
Timecounter "TSC-low" frequency 1446761464 Hz quality 1000
Root mount waiting for: usbus1 usbus0
uhub0: 12 ports with 12 removable, self powered
Root mount waiting for: usbus1
Root mount waiting for: usbus1
Root mount waiting for: usbus1
Root mount waiting for: usbus1
uhub1: 12 ports with 12 removable, self powered
Trying to mount root from ufs:/dev/ufsid/57b1d01821cf2fd4 [rw]...
WARNING: / was not properly dismounted
<118>Configuring crash dumps...
<118>Using /dev/label/swap0 for dump device.
<118>** SU+J Recovering /dev/ufsid/57b1d01821cf2fd4
<118>** Reading 26279936 byte journal from inode 4.
<118>** Building recovery table.
<118>** Resolving unreferenced inode list.
<118>** Processing journal entries.

<118>** 66 journal records in 3584 bytes for 58.93% utilization
<118>** Freed 6 inodes (0 dirs) 2 blocks, and 5 frags.
<118>
<118>***** FILE SYSTEM MARKED CLEAN *****
<118>Filesystems are clean, continuing...
<118>Mounting filesystems...
<118>
<118>   ___
<118> ___/ f \
<118>/ p ___/ Sense
<118>___/   \
<118>   ___/
<118>
<118>Welcome to pfSense 2.3.2-RELEASE on the 'pfSense' platform...
<118>
<118>savecore: reboot after panic: ffs_valloc: dup alloc
<118>savecore: writing core to /var/crash/textdump.tar.0
<118>Creating symlinks.....ELF ldconfig path: /lib /usr/lib /usr/lib/compat /usr/local/lib /
usr/local/lib/ipsec /usr/local/lib/perl5/5.20/mach/CORE
<118>a.out ldconfig path: /usr/lib/aout /usr/lib/compat/aout
<118>done.
<118>pkg: sqlite error while executing INSERT INTO pkg_lock_pid VALUES (?1); in file
pkgdb.c:2674: UNIQUE constraint failed: pkg_lock_pid.pid
<118>pkg: Cannot get an exclusive lock on a database, it is locked by another process
mode = 0100666, inum = 17, fs = /
panic: ffs_valloc: dup alloc
cpuid = 0
KDB: enter: panic
panic.txt06000025127547641367154 ustarrootwheelffs_valloc: dup
allocversion.txt060000267127547641367637 ustarrootwheelFreeBSD 10.3-RELEASE-
p5 #0 7307492(RELENG_2_3_2): Tue Jul 19 13:53:47 CDT 2016
    root@ce23-i386-builder:/builder/pfsense-232/tmp/obj/builder/pfsense-232/tmp/
FreeBSD-src/sys/pfSense

Filename: /var/crash/textdump.tar.last
ddb.txt06000013657112754764136 7133 ustarrootwheeldb:0:kdb.enter.default> run
lockinfo
db:1:lockinfo> show locks
No such command
db:1:locks> show alllocks
No such command
db:1:alllocks> show lockedvnods
Locked vnodes
db:0:kdb.enter.default> show pcpu
cpuid   = 0
dynamic pcpu = 0x56f000
curthread   = 0xc6f4cc80: pid 149 "php-cgi"

```
curpcb       = 0xe9af6c00
fpcurthread  = none
idlethread   = 0xc6716000: tid 100003 "idle: cpu0"
APIC ID      = 0
currentldt   = 0x50
db:0:kdb.enter.default>  bt
Tracing pid 149 tid 100069 td 0xc6f4cc80
kdb_enter(c147c5b6,c147c5b6,c14afaab,e9af667c,0,...) at kdb_enter+0x3d/frame
0xe9af6628
vpanic(c14afaab,e9af667c,e9af667c,e9af66f8,c10007b1,...) at vpanic+0x13b/frame
0xe9af665c
panic(c14afaab,81b6,11,c6dc00d4,8180,...) at panic+0x1b/frame 0xe9af6670
ffs_valloc(c6edfd50,8180,c6dbcc00,e9af673c,0,...) at ffs_valloc+0x961/frame
0xe9af66f8
ufs_makeinode(e9af6a50,e9af6a64,c6edfd50,e9af6a08,e9af68c8,...) at
ufs_makeinode+0x73/frame 0xe9af6878
ufs_create(e9af6970,2,c6f4cc80,c1eb5990,c6edfd50,...) at ufs_create+0x30/frame
0xe9af689c
VOP_CREATE_APV(c1ea0728,e9af6970,2,c6edfd50,0,...) at
VOP_CREATE_APV+0x95/frame 0xe9af68c8
vn_open_cred(e9af6a08,e9af6a94,180,0,c6dbcc00,c6dde4d0) at vn_open_cred+0x2ae/
frame 0xe9af6998
vn_open(e9af6a08,e9af6a94,180,c6dde4d0,287e5e28,...) at vn_open+0x3d/frame
0xe9af69c0
kern_openat(c6f4cc80,ffffff9c,287e5e28,0,a02,180) at kern_openat+0x310/frame
0xe9af6ab4
sys_open(c6f4cc80,e9af6b68,c6f48c20,c0d1bba2,c6f4cc80,...) at sys_open+0x39/
frame 0xe9af6ad8
syscall(e9af6ba8) at syscall+0x5c9/frame 0xe9af6b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9af6b9c
--- syscall (5, FreeBSD ELF32, sys_open), eip = 0x287447af, esp = 0xbfbfe06c, ebp =
0xbfbfe514 ---
db:0:kdb.enter.default>  ps
  pid  ppid  pgrp   uid   state   wmesg     wchan    cmd
  150   148    21     0  S+      piperd   0xc6dee000 grep
  149   148    21     0  R+      CPU 0                php-cgi
  148   136    21     0  S+      wait     0xc6f49610 sh
  136    21    21     0  S+      piperd   0xc6dee198 sh
   55     0     0     0  DL      mdwait   0xc6d7d800 [md0]
   21     1    21     0  Ss+     pause    0xc6d8a97c sh
   20     0     0     0  DL      vlruwt   0xc6d8ac20 [vnlru]
   19     0     0     0  DL      syncer   0xc1f99fc4 [syncer]
   18     0     0     0  DL      (threaded)          [bufdaemon]
100061                  D       sdflush  0xc6914284 [/ worker]
100054                  D       psleep   0xc1f99704 [bufdaemon]
   17     0     0     0  DL      pgzero   0xc1fa41a8 [pagezero]
    9     0     0     0  DL      pollid   0xc1f797f0 [idlepoll]
```

```
    8     0     0     0  DL      psleep   0xc1fa3ec4 [vmdaemon]
    7     0     0     0  DL      (threaded)          [pagedaemon]
100057                   D       umarcl   0xc1fa3aa4 [uma]
100050                   D       psleep   0xc2027604 [pagedaemon]
    6     0     0     0  DL      waiting_ 0xc2021f74 [sctp_iterator]
    5     0     0     0  DL      pftm     0xc0f94ba0 [pf purge]
   16     0     0     0  DL      (threaded)          [usb]
100041                   D       -        0xc6893d64 [usbus1]
100040                   D       -        0xc6893d34 [usbus1]
100039                   D       -        0xc6893d04 [usbus1]
100038                   D       -        0xc6893cd4 [usbus1]
100037                   D       -        0xc6893ca4 [usbus1]
100033                   D       -        0xc67fcb8c [usbus0]
100032                   D       -        0xc67fcb5c [usbus0]
100031                   D       -        0xc67fcb2c [usbus0]
100030                   D       -        0xc67fcafc [usbus0]
100029                   D       -        0xc67fcacc [usbus0]
    4     0     0     0  DL      (threaded)          [cam]
100049                   D       -        0xc1eb8eac [scanner]
100017                   D       -        0xc1eb9000 [doneq0]
   15     0     0     0  DL      -        0xc1ed60c8 [rand_harvestq]
    3     0     0     0  DL      crypto_r 0xc1fa2e28 [crypto returns]
    2     0     0     0  DL      crypto_w 0xc1fa2d68 [crypto]
   14     0     0     0  DL      (threaded)          [geom]
100010                   D       -        0xc201cfd8 [g_down]
100009                   D       -        0xc201cfd4 [g_up]
100008                   D       -        0xc201cfd0 [g_event]
   13     0     0     0  DL      sleep    0xc1e9081c [ng_queue0]
   12     0     0     0  WL      (threaded)          [intr]
100047                   I                         [swi1: pfsync]
100045                   I                         [swi1: pf send]
100043                   I                         [irq12: psm0]
100042                   I                         [irq1: atkbd0]
100036                   I                         [irq10: em0 ehci0]
100028                   I                         [irq11: pcm0 em2+]
100026                   I                         [irq15: ata1]
100025                   I                         [irq14: ata0]
100024                   I                         [swi6: Giant taskq]
100022                   I                         [swi6: task queue]
100016                   I                         [swi5: fast taskq]
100006                   I                         [swi4: clock]
100005                   I                         [swi1: netisr 0]
100004                   I                         [swi3: vm]
   11     0     0     0  RL                         [idle: cpu0]
    1     0     1     0  SLs     wait     0xc670f308 [init]
   10     0     0     0  DL      audit_wo 0xc2025edc [audit]
    0     0     0     0  DLs     (threaded)          [kernel]
```

```
100048                D      -       0xc6712980 [CAM taskq]
100035                D      -       0xc68d0480 [em2 taskq]
100034                D      -       0xc68d0700 [em1 taskq]
100027                D      -       0xc6883300 [em0 taskq]
100023                D      -       0xc6712400 [ffs_trim taskq]
100021                D      -       0xc6712680 [kqueue taskq]
100020                D      -       0xc6712700 [acpi_task_2]
100019                D      -       0xc6712700 [acpi_task_1]
100018                D      -       0xc6712700 [acpi_task_0]
100015                D      -       0xc6712b00 [thread taskq]
100011                D      -       0xc6713e80 [firmware taskq]
100000                D      swapin   0xc201d054 [swapper]
db:0:kdb.enter.default>  alltrace

Tracing command grep pid 150 tid 100067 td 0xc6f4d320
sched_switch(c6f4d320,0,104,ac,98,...) at sched_switch+0x2d2/frame 0xe9af089c
mi_switch(104,0,c6f4d320,e9af094c,c0cf40f2,c6f4d320) at mi_switch+0x122/frame
0xe9af08d4
sleepq_switch(c6f4d320,0,c1480b9a,1a8,e9af0960,...) at sleepq_switch+0x15b/frame
0xe9af08fc
sleepq_catch_signals(0,c1480b9a,154,0,0,...) at sleepq_catch_signals+0x5be/frame
0xe9af0950
sleepq_wait_sig(c6dee000,5c,c1481733,100,0,...) at sleepq_wait_sig+0x14/frame
0xe9af097c
_sleep(c6dee000,c6dee180,15c,c1481733,0,...) at _sleep+0x29b/frame 0xe9af09c4
pipe_read(c6dde658,e9af0aa8,c6dbcc00,0,c6f4d320,...) at pipe_read+0x31a/frame
0xe9af0a08
dofileread(c6dde658,e9af0aa8,ffffffff,ffffffff,0,...) at dofileread+0x9e/frame 0xe9af0a40
kern_readv(c6f4d320,0,e9af0aa8,e9af0ad8,2881c000,...) at kern_readv+0x96/frame
0xe9af0a88
sys_read(c6f4d320,e9af0b68,c6f49308,c6f4d320,c6f49308,...) at sys_read+0x5c/frame
0xe9af0ad8
syscall(e9af0ba8) at syscall+0x5c9/frame 0xe9af0b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9af0b9c
--- syscall (3, FreeBSD ELF32, sys_read), eip = 0x281de85f, esp = 0xbfbfec18, ebp =
0xbfbfec60 ---

Tracing command php-cgi pid 149 tid 100069 td 0xc6f4cc80
kdb_enter(c147c5b6,c147c5b6,c14afaab,e9af667c,0,...) at kdb_enter+0x3d/frame
0xe9af6628
vpanic(c14afaab,e9af667c,e9af667c,e9af66f8,c10007b1,...) at vpanic+0x13b/frame
0xe9af665c
panic(c14afaab,81b6,11,c6dc00d4,8180,...) at panic+0x1b/frame 0xe9af6670
ffs_valloc(c6edfd50,8180,c6dbcc00,e9af673c,0,...) at ffs_valloc+0x961/frame
0xe9af66f8
ufs_makeinode(e9af6a50,e9af6a64,c6edfd50,e9af6a08,e9af68c8,...) at
ufs_makeinode+0x73/frame 0xe9af6878
```

ufs_create(e9af6970,2,c6f4cc80,c1eb5990,c6edfd50,...) at ufs_create+0x30/frame 0xe9af689c
VOP_CREATE_APV(c1ea0728,e9af6970,2,c6edfd50,0,...) at VOP_CREATE_APV+0x95/frame 0xe9af68c8
vn_open_cred(e9af6a08,e9af6a94,180,0,c6dbcc00,c6dde4d0) at vn_open_cred+0x2ae/frame 0xe9af6998
vn_open(e9af6a08,e9af6a94,180,c6dde4d0,287e5e28,...) at vn_open+0x3d/frame 0xe9af69c0
kern_openat(c6f4cc80,ffffff9c,287e5e28,0,a02,180) at kern_openat+0x310/frame 0xe9af6ab4
sys_open(c6f4cc80,e9af6b68,c6f48c20,c0d1bba2,c6f4cc80,...) at sys_open+0x39/frame 0xe9af6ad8
syscall(e9af6ba8) at syscall+0x5c9/frame 0xe9af6b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9af6b9c
--- syscall (5, FreeBSD ELF32, sys_open), eip = 0x287447af, esp = 0xbfbfe06c, ebp = 0xbfbfe514 ---

Tracing command sh pid 148 tid 100066 td 0xc6f4d640
sched_switch(c6f4d640,0,104,ac,98,...) at sched_switch+0x2d2/frame 0xe9aed820
mi_switch(104,0,e9aed8d0,c0cf40f2,c1f86e80,c6f4d640) at mi_switch+0x122/frame 0xe9aed858
sleepq_switch(c6f4d640,0,c1480b9a,1a8,0,...) at sleepq_switch+0x15b/frame 0xe9aed880
sleepq_catch_signals(0,c1480b9a,154,0,3457e5ce,...) at sleepq_catch_signals+0x5be/frame 0xe9aed8d4
sleepq_wait_sig(c6f49610,6c,c1481bd3,100,0,...) at sleepq_wait_sig+0x14/frame 0xe9aed900
_sleep(c6f49610,c6f496a8,16c,c1481bd3,0,...) at _sleep+0x29b/frame 0xe9aed948
kern_wait6(c6f4d640,7,0,0,e9aed9e4,...) at kern_wait6+0x71a/frame 0xe9aed9b8
sys_wait4(c6f4d640,e9aedb68,c6f49610,c6f4d640,c6f49610,...) at sys_wait4+0x94/frame 0xe9aedad8
syscall(e9aedba8) at syscall+0x5c9/frame 0xe9aedb9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9aedb9c
--- syscall (7, FreeBSD ELF32, sys_wait4), eip = 0x281acf43, esp = 0xbfbfe918, ebp = 0xbfbfe930 ---

Tracing command sh pid 136 tid 100062 td 0xc6d96000
sched_switch(c6d96000,0,104,ac,98,...) at sched_switch+0x2d2/frame 0xe9ade89c
mi_switch(104,0,c6d96000,e9ade94c,c0cf40f2,c6d96000) at mi_switch+0x122/frame 0xe9ade8d4
sleepq_switch(c6d96000,0,c1480b9a,1a8,c27a284c,...) at sleepq_switch+0x15b/frame 0xe9ade8fc
sleepq_catch_signals(0,c1480b9a,154,0,0,...) at sleepq_catch_signals+0x5be/frame 0xe9ade950
sleepq_wait_sig(c6dee198,5c,c1481733,100,0,...) at sleepq_wait_sig+0x14/frame 0xe9ade97c
_sleep(c6dee198,c6dee318,15c,c1481733,0,...) at _sleep+0x29b/frame 0xe9ade9c4

pipe_read(c6dde5b0,e9adeaa8,c6dbcc00,0,c6d96000,...) at pipe_read+0x31a/frame 0xe9adea08
dofileread(c6dde5b0,e9adeaa8,ffffffff,ffffffff,0,...) at dofileread+0x9e/frame 0xe9adea40
kern_readv(c6d96000,3,e9adeaa8,e9adead8,bfbfeb60,...) at kern_readv+0x96/frame 0xe9adea88
sys_read(c6d96000,e9adeb68,c6d8a000,0,0,...) at sys_read+0x5c/frame 0xe9adead8
syscall(e9adeba8) at syscall+0x5c9/frame 0xe9adeb9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9adeb9c
--- syscall (3, FreeBSD ELF32, sys_read), eip = 0x2820f85f, esp = 0xbfbfeacc, ebp = 0xbfbfebf4 ---

Tracing command md0 pid 55 tid 100064 td 0xc690c960
sched_switch(c690c960,0,104,e9ae6a70,0,...) at sched_switch+0x2d2/frame 0xe9ae6a48
mi_switch(104,0,da8c2370,c6e02100,c690c960,c6d7d800) at mi_switch+0x122/frame 0xe9ae6a80
sleepq_switch(c690c960,0,c1480b9a,269,c6dbc3d8,...) at sleepq_switch+0x15b/frame 0xe9ae6aa8
sleepq_wait(c6d7d800,5c,c1401bd9,0,0,...) at sleepq_wait+0x3f/frame 0xe9ae6ad4
_sleep(c6d7d800,c6d7d820,25c,c1401bd9,0,...) at _sleep+0x2ae/frame 0xe9ae6b1c
md_kthread(c6d7d800,e9ae6ba8,c1eb1abc,2880a4a0,e9ae6b98,...) at md_kthread+0xd7/frame 0xe9ae6b64
fork_exit(c08726b0,c6d7d800,e9ae6ba8) at fork_exit+0xa3/frame 0xe9ae6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xe9ae6b94
--- trap 0, eip = 0, esp = 0xe9ae6be0, ebp = 0 ---

Tracing command sh pid 21 tid 100058 td 0xc6d96c80
sched_switch(c6d96c80,0,104,ac,98,...) at sched_switch+0x2d2/frame 0xe9ad093c
mi_switch(104,0,c6d96c80,e9ad09ec,c0cf40f2,c6d96c80) at mi_switch+0x122/frame 0xe9ad0974
sleepq_switch(c6d96c80,0,c1480b9a,1a8,0,...) at sleepq_switch+0x15b/frame 0xe9ad099c
sleepq_catch_signals(0,c1480b9a,154,c6d96c80,e9ad0a64,...) at sleepq_catch_signals+0x5be/frame 0xe9ad09f0
sleepq_wait_sig(c6d8a97c,74,c1418125,100,0,...) at sleepq_wait_sig+0x14/frame 0xe9ad0a1c
_sleep(c6d8a97c,c6d8a9b0,174,c1418125,0,...) at _sleep+0x29b/frame 0xe9ad0a64
kern_sigsuspend(c6d96c80,0,0,0,0,...) at kern_sigsuspend+0x137/frame 0xe9ad0aa8
sys_sigsuspend(c6d96c80,e9ad0b68,c6d8a918,0,0,...) at sys_sigsuspend+0x58/frame 0xe9ad0ad8
syscall(e9ad0ba8) at syscall+0x5c9/frame 0xe9ad0b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xe9ad0b9c
--- syscall (4, FreeBSD ELF32, sys_write), eip = 0x281acf83, esp = 0xbfbfec0c, ebp = 0xbfbfec98 ---

Tracing command vnlru pid 20 tid 100056 td 0xc6d97320
sched_switch(c6d97320,0,104,0,c201e5a4,...) at sched_switch+0x2d2/frame

0xdb6a2a1c
mi_switch(104,0,101,c6d97320,db6a2aa8,c6d8ac20) at mi_switch+0x122/frame 0xdb6a2a54
sleepq_switch(c6d97320,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdb6a2a7c
sleepq_timedwait(c6d8ac20,60,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdb6a2aa8
_sleep(c6d8ac20,c1f99f7c,260,c1487ef8,ffffffed8,...) at _sleep+0x282/frame 0xdb6a2af0
vnlru_proc(0,db6a2ba8,0,0,0,...) at vnlru_proc+0xcf/frame 0xdb6a2b64
fork_exit(c0dc7f10,0,db6a2ba8) at fork_exit+0xa3/frame 0xdb6a2b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdb6a2b94
--- trap 0, eip = 0, esp = 0xdb6a2be0, ebp = 0 ---

Tracing command syncer pid 19 tid 100055 td 0xc6d97640
sched_switch(c6d97640,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xdb69fa28
mi_switch(104,0,101,c6d97640,db69fab4,c1f99fc4) at mi_switch+0x122/frame 0xdb69fa60
sleepq_switch(c6d97640,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdb69fa88
sleepq_timedwait(c1f99fc4,0,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdb69fab4
_cv_timedwait_sbt(c1f99fc4,c1f99fa4,ffffffed8,0,0,...) at _cv_timedwait_sbt+0x1b7/frame 0xdb69fae8
sched_sync(0,db69fba8,0,0,0,...) at sched_sync+0x7a1/frame 0xdb69fb64
fork_exit(c0dc75d0,0,db69fba8) at fork_exit+0xa3/frame 0xdb69fb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdb69fb94
--- trap 0, eip = 0, esp = 0xdb69fbe0, ebp = 0 ---

Tracing command bufdaemon pid 18 tid 100061 td 0xc6d96320
sched_switch(c6d96320,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xe9adba50
mi_switch(104,0,101,c6d96320,e9adbadc,c6914284) at mi_switch+0x122/frame 0xe9adba88
sleepq_switch(c6d96320,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xe9adbab0
sleepq_timedwait(c6914284,54,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xe9adbadc
_sleep(c6914284,c6914200,54,c14b2d06,7fffff6c,...) at _sleep+0x282/frame 0xe9adbb24
softdep_flush(c6db4d20,e9adbba8,c10178f0,0,e9adbb90,...) at softdep_flush+0x1fa/frame 0xe9adbb64
fork_exit(c10178f0,c6db4d20,e9adbba8) at fork_exit+0xa3/frame 0xe9adbb94
fork_trampoline() at fork_trampoline+0x8/frame 0xe9adbb94
--- trap 0, eip = 0, esp = 0xe9adbbe0, ebp = 0 ---

Tracing command bufdaemon pid 18 tid 100054 td 0xc6d97960
sched_switch(c6d97960,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xdb69ca58
mi_switch(104,0,101,c6d97960,db69cae4,c1f99704) at mi_switch+0x122/frame

0xdb69ca90
sleepq_switch(c6d97960,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdb69cab8
sleepq_timedwait(c1f99704,54,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdb69cae4
_sleep(c1f99704,c1f99680,54,c148589d,fffffed8,...) at _sleep+0x282/frame 0xdb69cb2c
buf_daemon(0,db69cba8,0,0,0,...) at buf_daemon+0xac/frame 0xdb69cb64
fork_exit(c0dab150,0,db69cba8) at fork_exit+0xa3/frame 0xdb69cb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdb69cb94
--- trap 0, eip = 0, esp = 0xdb69cbe0, ebp = 0 ---

Tracing command pagezero pid 17 tid 100053 td 0xc6d97c80
sched_switch(c6d97c80,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame 0xdb699a60
mi_switch(104,0,101,c6d97c80,db699aec,c1fa41a8) at mi_switch+0x122/frame 0xdb699a98
sleepq_switch(c6d97c80,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdb699ac0
sleepq_timedwait(c1fa41a8,0,12b,0,0,...) at sleepq_timedwait+0x3f/frame 0xdb699aec
_sleep(c1fa41a8,c2027580,0,c14b71b2,fffea520,...) at _sleep+0x282/frame 0xdb699b34
vm_pagezero(0,db699ba8,0,0,0,...) at vm_pagezero+0xd2/frame 0xdb699b64
fork_exit(c107ad10,0,db699ba8) at fork_exit+0xa3/frame 0xdb699b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdb699b94
--- trap 0, eip = 0, esp = 0xdb699be0, ebp = 0 ---

Tracing command idlepoll pid 9 tid 100052 td 0xc68b9c80
sched_switch(c68b9c80,0,104,0,c201e5a4,...) at sched_switch+0x2d2/frame 0xdabffa4c
mi_switch(104,0,101,c68b9c80,dabffad8,c1f797f0) at mi_switch+0x122/frame 0xdabffa84
sleepq_switch(c68b9c80,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdabffaac
sleepq_timedwait(c1f797f0,0,2,0,0,...) at sleepq_timedwait+0x3f/frame 0xdabffad8
_sleep(c1f797f0,0,0,c147ae9d,fffffc88,...) at _sleep+0x282/frame 0xdabffb20
poll_idle(0,dabffba8,0,0,0,...) at poll_idle+0x125/frame 0xdabffb64
fork_exit(c0cf7de0,0,dabffba8) at fork_exit+0xa3/frame 0xdabffb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabffb94
--- trap 0, eip = 0, esp = 0xdabffbe0, ebp = 0 ---

Tracing command vmdaemon pid 8 tid 100051 td 0xc690a000
sched_switch(c690a000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabfca14
mi_switch(104,0,0,0,c690a000,c1fa3ec4) at mi_switch+0x122/frame 0xdabfca4c
sleepq_switch(c690a000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdabfca74
sleepq_wait(c1fa3ec4,74,c148589d,0,0,...) at sleepq_wait+0x3f/frame 0xdabfcaa0
_sleep(c1fa3ec4,c1fa3eac,74,c148589d,0,...) at _sleep+0x2ae/frame 0xdabfcae8
vm_daemon(0,dabfcba8,0,0,0,...) at vm_daemon+0x130/frame 0xdabfcb64

fork_exit(c1072d80,0,dabfcba8) at fork_exit+0xa3/frame 0xdabfcb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabfcb94
--- trap 0, eip = 0, esp = 0xdabfcbe0, ebp = 0 ---

Tracing command pagedaemon pid 7 tid 100057 td 0xc6d97000
sched_switch(c6d97000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xe9acca64
mi_switch(104,0,0,0,c6d97000,c1fa3aa4) at mi_switch+0x122/frame 0xe9acca9c
sleepq_switch(c6d97000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xe9accac4
sleepq_wait(c1fa3aa4,54,c14b4e9f,0,0,...) at sleepq_wait+0x3f/frame 0xe9accaf0
_sleep(c1fa3aa4,c1fa3a90,54,c14b4e9f,0,...) at _sleep+0x2ae/frame 0xe9accb38
uma_reclaim_worker(0,e9accba8,0,0,0,...) at uma_reclaim_worker+0xb0/frame
0xe9accb64
fork_exit(c10545c0,0,e9accba8) at fork_exit+0xa3/frame 0xe9accb94
fork_trampoline() at fork_trampoline+0x8/frame 0xe9accb94
--- trap 0, eip = 0, esp = 0xe9accbe0, ebp = 0 ---

Tracing command pagedaemon pid 7 tid 100050 td 0xc690a320
sched_switch(c690a320,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame
0xdabf9a20
mi_switch(104,0,101,c690a320,dabf9aac,c2027604) at mi_switch+0x122/frame
0xdabf9a58
sleepq_switch(c690a320,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame
0xdabf9a80
sleepq_timedwait(c2027604,54,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdabf9aac
_sleep(c2027604,c2027580,54,c148589d,fffffed8,...) at _sleep+0x282/frame 0xdabf9af4
vm_pageout(0,dabf9ba8,0,0,0,...) at vm_pageout+0x23b/frame 0xdabf9b64
fork_exit(c1073b20,0,dabf9ba8) at fork_exit+0xa3/frame 0xdabf9b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabf9b94
--- trap 0, eip = 0, esp = 0xdabf9be0, ebp = 0 ---

Tracing command sctp_iterator pid 6 tid 100046 td 0xc690b000
sched_switch(c690b000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabeda64
mi_switch(104,0,0,0,c690b000,c2021f74) at mi_switch+0x122/frame 0xdabeda9c
sleepq_switch(c690b000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdabedac4
sleepq_wait(c2021f74,0,c149c033,0,0,...) at sleepq_wait+0x3f/frame 0xdabedaf0
_sleep(c2021f74,c2021f3c,0,c149c033,0,...) at _sleep+0x2ae/frame 0xdabedb38
sctp_iterator_thread(0,dabedba8,0,0,0,...) at sctp_iterator_thread+0x9c/frame
0xdabedb64
fork_exit(c0ea87d0,0,dabedba8) at fork_exit+0xa3/frame 0xdabedb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabedb94
--- trap 0, eip = 0, esp = 0xdabedbe0, ebp = 0 ---

Tracing command pf purge pid 5 tid 100044 td 0xc690b640
sched_switch(c690b640,0,104,c201e5a4,0,...) at sched_switch+0x2d2/frame
0xdabe6a58

mi_switch(104,0,101,c690b640,dabe6ae4,c0f94ba0) at mi_switch+0x122/frame 0xdabe6a90
sleepq_switch(c690b640,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame 0xdabe6ab8
sleepq_timedwait(c0f94ba0,0,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdabe6ae4
_sleep(c0f94ba0,c2024ef4,0,c14a8087,1999997c,...) at _sleep+0x282/frame 0xdabe6b2c
pf_purge_thread(0,dabe6ba8,0,0,0,...) at pf_purge_thread+0xba/frame 0xdabe6b64
fork_exit(c0f94ba0,0,dabe6ba8) at fork_exit+0xa3/frame 0xdabe6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabe6b94
--- trap 0, eip = 0, esp = 0xdabe6be0, ebp = 0 ---

Tracing command usb pid 16 tid 100041 td 0xc690c000
sched_switch(c690c000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabdda78
mi_switch(104,0,0,0,c690c000,c6893d64) at mi_switch+0x122/frame 0xdabddab0
sleepq_switch(c690c000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdabddad8
sleepq_wait(c6893d64,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabddb04
_cv_wait(c6893d64,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabddb34
usb_process(c6893d5c,dabddba8,0,0,0,...) at usb_process+0xf6/frame 0xdabddb64
fork_exit(c0ab1850,c6893d5c,dabddba8) at fork_exit+0xa3/frame 0xdabddb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabddb94
--- trap 0, eip = 0, esp = 0xdabddbe0, ebp = 0 ---

Tracing command usb pid 16 tid 100040 td 0xc690c320
sched_switch(c690c320,0,104,4000c,0,...) at sched_switch+0x2d2/frame 0xdabdaa78
mi_switch(104,0,c68d2fc8,c0d1ba1d,c690c320,c6893d34) at mi_switch+0x122/frame 0xdabdaab0
sleepq_switch(c690c320,0,c1480b9a,269,500,...) at sleepq_switch+0x15b/frame 0xdabdaad8
sleepq_wait(c6893d34,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabdab04
_cv_wait(c6893d34,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabdab34
usb_process(c6893d2c,dabdaba8,0,0,0,...) at usb_process+0xf6/frame 0xdabdab64
fork_exit(c0ab1850,c6893d2c,dabdaba8) at fork_exit+0xa3/frame 0xdabdab94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabdab94
--- trap 0, eip = 0, esp = 0xdabdabe0, ebp = 0 ---

Tracing command usb pid 16 tid 100039 td 0xc6857320
sched_switch(c6857320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabd7a78
mi_switch(104,0,0,0,c6857320,c6893d04) at mi_switch+0x122/frame 0xdabd7ab0
sleepq_switch(c6857320,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdabd7ad8
sleepq_wait(c6893d04,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabd7b04
_cv_wait(c6893d04,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabd7b34
usb_process(c6893cfc,dabd7ba8,0,0,0,...) at usb_process+0xf6/frame 0xdabd7b64
fork_exit(c0ab1850,c6893cfc,dabd7ba8) at fork_exit+0xa3/frame 0xdabd7b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabd7b94

--- trap 0, eip = 0, esp = 0xdabd7be0, ebp = 0 ---

Tracing command usb pid 16 tid 100038 td 0xc6857640
sched_switch(c6857640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabd4a78
mi_switch(104,0,0,0,c6857640,c6893cd4) at mi_switch+0x122/frame 0xdabd4ab0
sleepq_switch(c6857640,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdabd4ad8
sleepq_wait(c6893cd4,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabd4b04
_cv_wait(c6893cd4,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabd4b34
usb_process(c6893ccc,dabd4ba8,0,0,0,...) at usb_process+0xf6/frame 0xdabd4b64
fork_exit(c0ab1850,c6893ccc,dabd4ba8) at fork_exit+0xa3/frame 0xdabd4b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabd4b94
--- trap 0, eip = 0, esp = 0xdabd4be0, ebp = 0 ---

Tracing command usb pid 16 tid 100037 td 0xc6857960
sched_switch(c6857960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabd1a78
mi_switch(104,0,0,0,c6857960,c6893ca4) at mi_switch+0x122/frame 0xdabd1ab0
sleepq_switch(c6857960,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdabd1ad8
sleepq_wait(c6893ca4,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdabd1b04
_cv_wait(c6893ca4,c6893ed0,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdabd1b34
usb_process(c6893c9c,dabd1ba8,0,0,0,...) at usb_process+0xf6/frame 0xdabd1b64
fork_exit(c0ab1850,c6893c9c,dabd1ba8) at fork_exit+0xa3/frame 0xdabd1b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabd1b94
--- trap 0, eip = 0, esp = 0xdabd1be0, ebp = 0 ---

Tracing command usb pid 16 tid 100033 td 0xc68b4640
sched_switch(c68b4640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab83a78
mi_switch(104,0,0,0,c68b4640,c67fcb8c) at mi_switch+0x122/frame 0xdab83ab0
sleepq_switch(c68b4640,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdab83ad8
sleepq_wait(c67fcb8c,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab83b04
_cv_wait(c67fcb8c,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab83b34
usb_process(c67fcb84,dab83ba8,0,0,0,...) at usb_process+0xf6/frame 0xdab83b64
fork_exit(c0ab1850,c67fcb84,dab83ba8) at fork_exit+0xa3/frame 0xdab83b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab83b94
--- trap 0, eip = 0, esp = 0xdab83be0, ebp = 0 ---

Tracing command usb pid 16 tid 100032 td 0xc68b4960
sched_switch(c68b4960,0,104,4000c,0,...) at sched_switch+0x2d2/frame 0xdab80a78
mi_switch(104,0,c68b0fc8,c0d1ba1d,c68b4960,c67fcb5c) at mi_switch+0x122/frame
0xdab80ab0
sleepq_switch(c68b4960,0,c1480b9a,269,100,...) at sleepq_switch+0x15b/frame
0xdab80ad8
sleepq_wait(c67fcb5c,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab80b04
_cv_wait(c67fcb5c,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab80b34
usb_process(c67fcb54,dab80ba8,0,0,0,...) at usb_process+0xf6/frame 0xdab80b64

fork_exit(c0ab1850,c67fcb54,dab80ba8) at fork_exit+0xa3/frame 0xdab80b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab80b94
--- trap 0, eip = 0, esp = 0xdab80be0, ebp = 0 ---

Tracing command usb pid 16 tid 100031 td 0xc68b4c80
sched_switch(c68b4c80,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab7da78
mi_switch(104,0,0,0,c68b4c80,c67fcb2c) at mi_switch+0x122/frame 0xdab7dab0
sleepq_switch(c68b4c80,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdab7dad8
sleepq_wait(c67fcb2c,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab7db04
_cv_wait(c67fcb2c,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab7db34
usb_process(c67fcb24,dab7dba8,0,0,0,...) at usb_process+0xf6/frame 0xdab7db64
fork_exit(c0ab1850,c67fcb24,dab7dba8) at fork_exit+0xa3/frame 0xdab7db94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab7db94
--- trap 0, eip = 0, esp = 0xdab7dbe0, ebp = 0 ---

Tracing command usb pid 16 tid 100030 td 0xc68b9000
sched_switch(c68b9000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab7aa78
mi_switch(104,0,0,0,c68b9000,c67fcafc) at mi_switch+0x122/frame 0xdab7aab0
sleepq_switch(c68b9000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdab7aad8
sleepq_wait(c67fcafc,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab7ab04
_cv_wait(c67fcafc,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab7ab34
usb_process(c67fcaf4,dab7aba8,0,0,0,...) at usb_process+0xf6/frame 0xdab7ab64
fork_exit(c0ab1850,c67fcaf4,dab7aba8) at fork_exit+0xa3/frame 0xdab7ab94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab7ab94
--- trap 0, eip = 0, esp = 0xdab7abe0, ebp = 0 ---

Tracing command usb pid 16 tid 100029 td 0xc68b9320
sched_switch(c68b9320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab77a78
mi_switch(104,0,0,0,c68b9320,c67fcacc) at mi_switch+0x122/frame 0xdab77ab0
sleepq_switch(c68b9320,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdab77ad8
sleepq_wait(c67fcacc,0,c1467e98,1,0,...) at sleepq_wait+0x3f/frame 0xdab77b04
_cv_wait(c67fcacc,c67fccf8,c14490f1,73,0,...) at _cv_wait+0x192/frame 0xdab77b34
usb_process(c67fcac4,dab77ba8,0,0,0,...) at usb_process+0xf6/frame 0xdab77b64
fork_exit(c0ab1850,c67fcac4,dab77ba8) at fork_exit+0xa3/frame 0xdab77b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab77b94
--- trap 0, eip = 0, esp = 0xdab77be0, ebp = 0 ---

Tracing command cam pid 4 tid 100049 td 0xc690a640
sched_switch(c690a640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabf6a3c
mi_switch(104,0,0,0,c690a640,c1eb8eac) at mi_switch+0x122/frame 0xdabf6a74
sleepq_switch(c690a640,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdabf6a9c
sleepq_wait(c1eb8eac,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdabf6ac8
_sleep(c1eb8eac,c1eb8efc,5c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdabf6b10

xpt_scanner_thread(0,dabf6ba8,0,0,0,...) at xpt_scanner_thread+0xcc/frame 0xdabf6b64
fork_exit(c04d0120,0,dabf6ba8) at fork_exit+0xa3/frame 0xdabf6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabf6b94
--- trap 0, eip = 0, esp = 0xdabf6be0, ebp = 0 ---

Tracing command cam pid 4 tid 100017 td 0xc6856640
sched_switch(c6856640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaae0a3c
mi_switch(104,0,0,daae0ad8,c6856640,c1eb9000) at mi_switch+0x122/frame 0xdaae0a74
sleepq_switch(c6856640,0,c1480b9a,269,c6d80000,...) at sleepq_switch+0x15b/frame 0xdaae0a9c
sleepq_wait(c1eb9000,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaae0ac8
_sleep(c1eb9000,c1eb8f80,5c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaae0b10
xpt_done_td(c1eb8f80,daae0ba8,0,0,0,...) at xpt_done_td+0xce/frame 0xdaae0b64
fork_exit(c04cfd00,c1eb8f80,daae0ba8) at fork_exit+0xa3/frame 0xdaae0b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaae0b94
--- trap 0, eip = 0, esp = 0xdaae0be0, ebp = 0 ---

Tracing command rand_harvestq pid 15 tid 100014 td 0xc6857000
sched_switch(c6857000,0,104,0,c201e5a4,...) at sched_switch+0x2d2/frame 0xdaad7a54
mi_switch(104,0,5,c6857000,daad7ae0,c1ed60c8) at mi_switch+0x122/frame 0xdaad7a8c
sleepq_switch(c6857000,0,c1480b9a,28c,5,...) at sleepq_switch+0x15b/frame 0xdaad7ab4
sleepq_timedwait(c1ed60c8,0,0,0,0,...) at sleepq_timedwait+0x3f/frame 0xdaad7ae0
msleep_spin_sbt(c1ed60c8,c2015ee0,c1467e98,19999999,0,...) at msleep_spin_sbt+0x1c0/frame 0xdaad7b18
random_kthread(c0989220,daad7ba8,0,0,0,...) at random_kthread+0x2a2/frame 0xdaad7b64
fork_exit(c09883d0,c0989220,daad7ba8) at fork_exit+0xa3/frame 0xdaad7b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaad7b94
--- trap 0, eip = 0, esp = 0xdaad7be0, ebp = 0 ---

Tracing command crypto returns pid 3 tid 100013 td 0xc6710000
sched_switch(c6710000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaad3a44
mi_switch(104,0,0,0,c6710000,c1fa2e28) at mi_switch+0x122/frame 0xdaad3a7c
sleepq_switch(c6710000,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdaad3aa4
sleepq_wait(c1fa2e28,6c,c14ac401,0,0,...) at sleepq_wait+0x3f/frame 0xdaad3ad0
_sleep(c1fa2e28,c1fa2e14,6c,c14ac401,0,...) at _sleep+0x2ae/frame 0xdaad3b18
crypto_ret_proc(0,daad3ba8,0,0,0,...) at crypto_ret_proc+0x20a/frame 0xdaad3b64
fork_exit(c0fd4c30,0,daad3ba8) at fork_exit+0xa3/frame 0xdaad3b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaad3b94
--- trap 0, eip = 0, esp = 0xdaad3be0, ebp = 0 ---

Tracing command crypto pid 2 tid 100012 td 0xc6710320
sched_switch(c6710320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaad0a50
mi_switch(104,0,0,0,c6710320,c1fa2d68) at mi_switch+0x122/frame 0xdaad0a88
sleepq_switch(c6710320,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdaad0ab0
sleepq_wait(c1fa2d68,6c,c14ac411,0,0,...) at sleepq_wait+0x3f/frame 0xdaad0adc
_sleep(c1fa2d68,c1fa2d50,6c,c14ac411,0,...) at _sleep+0x2ae/frame 0xdaad0b24
crypto_proc(0,daad0ba8,0,0,0,...) at crypto_proc+0x2e8/frame 0xdaad0b64
fork_exit(c0fd48e0,0,daad0ba8) at fork_exit+0xa3/frame 0xdaad0b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaad0b94
--- trap 0, eip = 0, esp = 0xdaad0be0, ebp = 0 ---

Tracing command geom pid 14 tid 100010 td 0xc6710960
sched_switch(c6710960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaacaa40
mi_switch(104,0,c6710960,daacaa98,c6710960,c201cfd8) at mi_switch+0x122/frame
0xdaacaa78
sleepq_switch(c6710960,0,c1480b9a,269,c6710960,...) at sleepq_switch+0x15b/frame
0xdaacaaa0
sleepq_wait(c201cfd8,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaacaacc
_sleep(c201cfd8,c1f76e88,25c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaacab14
g_io_schedule_down(c6710960,5c,c146ae6d,6b,daacab94,...) at
g_io_schedule_down+0x5c/frame 0xdaacab4c
g_down_procbody(0,daacaba8,0,0,0,...) at g_down_procbody+0x6d/frame 0xdaacab64
fork_exit(c0c56dd0,0,daacaba8) at fork_exit+0xa3/frame 0xdaacab94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaacab94
--- trap 0, eip = 0, esp = 0xdaacabe0, ebp = 0 ---

Tracing command geom pid 14 tid 100009 td 0xc6710c80
sched_switch(c6710c80,0,104,c65b7e40,0,...) at sched_switch+0x2d2/frame
0xdaac7a40
mi_switch(104,0,8,c6710c80,c6710c80,c201cfd4) at mi_switch+0x122/frame
0xdaac7a78
sleepq_switch(c6710c80,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdaac7aa0
sleepq_wait(c201cfd4,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaac7acc
_sleep(c201cfd4,c1f76ea8,25c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaac7b14
g_io_schedule_up(c6710c80,5c,c146ae6d,5e,daac7b94,...) at g_io_schedule_up+0xd5/
frame 0xdaac7b4c
g_up_procbody(0,daac7ba8,0,0,0,...) at g_up_procbody+0x6d/frame 0xdaac7b64
fork_exit(c0c56d60,0,daac7ba8) at fork_exit+0xa3/frame 0xdaac7b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaac7b94
--- trap 0, eip = 0, esp = 0xdaac7be0, ebp = 0 ---

Tracing command geom pid 14 tid 100008 td 0xc6711000
sched_switch(c6711000,0,104,0,c6d96c50,...) at sched_switch+0x2d2/frame
0xdaac4a54
mi_switch(104,0,c0d1bc92,c6d96960,c6711000,c201cfd0) at mi_switch+0x122/frame

0xdaac4a8c
sleepq_switch(c6711000,0,c1480b9a,269,daac4ad8,...) at sleepq_switch+0x15b/frame 0xdaac4ab4
sleepq_wait(c201cfd0,5c,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaac4ae0
_sleep(c201cfd0,c1f76e60,25c,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaac4b28
g_run_events(0,daac4ba8,0,0,0,...) at g_run_events+0x62/frame 0xdaac4b64
fork_exit(c0c56cf0,0,daac4ba8) at fork_exit+0xa3/frame 0xdaac4b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaac4b94
--- trap 0, eip = 0, esp = 0xdaac4be0, ebp = 0 ---

Tracing command ng_queue pid 13 tid 100007 td 0xc6711320
sched_switch(c6711320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaac1a50
mi_switch(104,0,0,0,c6711320,c1e9081c) at mi_switch+0x122/frame 0xdaac1a88
sleepq_switch(c6711320,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdaac1ab0
sleepq_wait(c1e9081c,8,c1485c29,0,0,...) at sleepq_wait+0x3f/frame 0xdaac1adc
_sleep(c1e9081c,c1f9cac0,8,c1485c29,0,...) at _sleep+0x2ae/frame 0xdaac1b24
ngthread(0,daac1ba8,0,0,0,...) at ngthread+0xbc/frame 0xdaac1b64
fork_exit(c0e55280,0,daac1ba8) at fork_exit+0xa3/frame 0xdaac1b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaac1b94
--- trap 0, eip = 0, esp = 0xdaac1be0, ebp = 0 ---

Tracing command intr pid 12 tid 100047 td 0xc690ac80
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100045 td 0xc690b320
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100043 td 0xc690b960
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100042 td 0xc690bc80
sched_switch(c690bc80,0,109,c1fa5808,c690bc80,...) at sched_switch+0x2d2/frame 0xdabe0af0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdabe0b28
ithread_loop(c68eced0,dabe0ba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdabe0b64
fork_exit(c0cd81d0,c68eced0,dabe0ba8) at fork_exit+0xa3/frame 0xdabe0b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabe0b94
--- trap 0, eip = 0, esp = 0xdabe0be0, ebp = 0 ---

Tracing command intr pid 12 tid 100036 td 0xc6857c80
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100028 td 0xc68b9640
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100026 td 0xc6716960

fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100025 td 0xc6716c80
sched_switch(c6716c80,0,109,c6854800,0,...) at sched_switch+0x2d2/frame 0xdab20af0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdab20b28
ithread_loop(c688a4a0,dab20ba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdab20b64
fork_exit(c0cd81d0,c688a4a0,dab20ba8) at fork_exit+0xa3/frame 0xdab20b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab20b94
--- trap 0, eip = 0, esp = 0xdab20be0, ebp = 0 ---

Tracing command intr pid 12 tid 100024 td 0xc6855000
sched_switch(c6855000,0,109,0,0,...) at sched_switch+0x2d2/frame 0xdaaf5af0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdaaf5b28
ithread_loop(c65bbd60,daaf5ba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdaaf5b64
fork_exit(c0cd81d0,c65bbd60,daaf5ba8) at fork_exit+0xa3/frame 0xdaaf5b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaf5b94
--- trap 0, eip = 0, esp = 0xdaaf5be0, ebp = 0 ---

Tracing command intr pid 12 tid 100022 td 0xc6855640
sched_switch(c6855640,0,109,0,0,...) at sched_switch+0x2d2/frame 0xdaaefaf0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdaaefb28
ithread_loop(c65bbd80,daaefba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdaaefb64
fork_exit(c0cd81d0,c65bbd80,daaefba8) at fork_exit+0xa3/frame 0xdaaefb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaefb94
--- trap 0, eip = 0, esp = 0xdaaefbe0, ebp = 0 ---

Tracing command intr pid 12 tid 100016 td 0xc6856960
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100006 td 0xc6711640
sched_switch(c6711640,0,109,c201e500,0,...) at sched_switch+0x2d2/frame 0xdaabeaf0
mi_switch(109,0,c14773bb,55b,0,...) at mi_switch+0x122/frame 0xdaabeb28
ithread_loop(c65ba0c0,daabeba8,0,0,0,...) at ithread_loop+0x1b1/frame 0xdaabeb64
fork_exit(c0cd81d0,c65ba0c0,daabeba8) at fork_exit+0xa3/frame 0xdaabeb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaabeb94
--- trap 0, eip = 0, esp = 0xdaabebe0, ebp = 0 ---

Tracing command intr pid 12 tid 100005 td 0xc6711960
fork_trampoline() at fork_trampoline

Tracing command intr pid 12 tid 100004 td 0xc6711c80
fork_trampoline() at fork_trampoline

Tracing command idle pid 11 tid 100003 td 0xc6716000
sched_switch(c6716000,0,608,0,56f000,...) at sched_switch+0x2d2/frame 0xdaab5a94

mi_switch(608,0,c147cb2c,d3,c1f86f04,...) at mi_switch+0x122/frame 0xdaab5acc
critical_exit(0,0,0,0,0,...) at critical_exit+0x89/frame 0xdaab5ae8
sched_idletd(0,daab5ba8,0,0,0,...) at sched_idletd+0x1dd/frame 0xdaab5b64
fork_exit(c0d3f8b0,0,daab5ba8) at fork_exit+0xa3/frame 0xdaab5b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaab5b94
--- trap 0, eip = 0, esp = 0xdaab5be0, ebp = 0 ---

Tracing command init pid 1 tid 100002 td 0xc6716320
sched_switch(c6716320,0,104,c105928c,0,...) at sched_switch+0x2d2/frame
0xdaab1820
mi_switch(104,0,daab18d0,c0cf40f2,c1f86e80,c6716320) at mi_switch+0x122/frame
0xdaab1858
sleepq_switch(c6716320,0,c1480b9a,1a8,0,...) at sleepq_switch+0x15b/frame
0xdaab1880
sleepq_catch_signals(0,c1480b9a,154,0,0,...) at sleepq_catch_signals+0x5be/frame
0xdaab18d4
sleepq_wait_sig(c670f308,6c,c1481bd3,100,0,...) at sleepq_wait_sig+0x14/frame
0xdaab1900
_sleep(c670f308,c670f3a0,16c,c1481bd3,0,...) at _sleep+0x29b/frame 0xdaab1948
kern_wait6(c6716320,7,0,0,daab19e4,...) at kern_wait6+0x71a/frame 0xdaab19b8
sys_wait4(c6716320,daab1b68,c670f308,c6716320,c1e73e50,...) at sys_wait4+0x94/
frame 0xdaab1ad8
syscall(daab1ba8) at syscall+0x5c9/frame 0xdaab1b9c
Xint0x80_syscall() at Xint0x80_syscall+0x2f/frame 0xdaab1b9c
--- syscall (7, FreeBSD ELF32, sys_wait4), eip = 0x8072c67, esp = 0xbfbfe8bc, ebp =
0xbfbfe8d0 ---

Tracing command audit pid 10 tid 100001 td 0xc6716640
sched_switch(c6716640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaaaea54
mi_switch(104,0,0,0,c6716640,c2025edc) at mi_switch+0x122/frame 0xdaaaea8c
sleepq_switch(c6716640,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame
0xdaaaeab4
sleepq_wait(c2025edc,0,c14ad51d,1,0,...) at sleepq_wait+0x3f/frame 0xdaaaeae0
_cv_wait(c2025edc,c2025e68,0,0,0,...) at _cv_wait+0x192/frame 0xdaaaeb10
audit_worker(0,daaaeba8,0,0,0,...) at audit_worker+0xa4/frame 0xdaaaeb64
fork_exit(c0ff7340,0,daaaeba8) at fork_exit+0xa3/frame 0xdaaaeb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaaeb94
--- trap 0, eip = 0, esp = 0xdaaaebe0, ebp = 0 ---

Tracing command kernel pid 0 tid 100048 td 0xc690a960
sched_switch(c690a960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabf3a58
mi_switch(104,0,dabf3ac8,c0d5f211,c690a960,c6712980) at mi_switch+0x122/frame
0xdabf3a90
sleepq_switch(c690a960,0,c1480b9a,269,c6712990,...) at sleepq_switch+0x15b/frame
0xdabf3ab8
sleepq_wait(c6712980,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdabf3ae4
_sleep(c6712980,c6712998,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdabf3b2c

taskqueue_thread_loop(c1eb8f24,dabf3ba8,0,0,0,...) at taskqueue_thread_loop+0x11b/
frame 0xdabf3b64
fork_exit(c0d63900,c1eb8f24,dabf3ba8) at fork_exit+0xa3/frame 0xdabf3b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabf3b94
--- trap 0, eip = 0, esp = 0xdabf3be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100035 td 0xc68b4000
sched_switch(c68b4000,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdabc9a68
mi_switch(104,0,0,0,c68b4000,c68d0480) at mi_switch+0x122/frame 0xdabc9aa0
sleepq_switch(c68b4000,0,c1480b9a,269,c68d0480,...) at sleepq_switch+0x15b/frame
0xdabc9ac8
sleepq_wait(c68d0480,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdabc9af4
msleep_spin_sbt(c68d0480,c68d0498,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/
frame 0xdabc9b2c
taskqueue_thread_loop(c68e35dc,dabc9ba8,0,0,0,...) at
taskqueue_thread_loop+0x15c/frame 0xdabc9b64
fork_exit(c0d63900,c68e35dc,dabc9ba8) at fork_exit+0xa3/frame 0xdabc9b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdabc9b94
--- trap 0, eip = 0, esp = 0xdabc9be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100034 td 0xc68b4320
sched_switch(c68b4320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaba6a68
mi_switch(104,0,0,0,c68b4320,c68d0700) at mi_switch+0x122/frame 0xdaba6aa0
sleepq_switch(c68b4320,0,c1480b9a,269,c68d0700,...) at sleepq_switch+0x15b/frame
0xdaba6ac8
sleepq_wait(c68d0700,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaba6af4
msleep_spin_sbt(c68d0700,c68d0718,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/
frame 0xdaba6b2c
taskqueue_thread_loop(c68cc5dc,daba6ba8,0,0,0,...) at
taskqueue_thread_loop+0x15c/frame 0xdaba6b64
fork_exit(c0d63900,c68cc5dc,daba6ba8) at fork_exit+0xa3/frame 0xdaba6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaba6b94
--- trap 0, eip = 0, esp = 0xdaba6be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100027 td 0xc68b9960
sched_switch(c68b9960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdab68a68
mi_switch(104,0,0,0,c68b9960,c6883300) at mi_switch+0x122/frame 0xdab68aa0
sleepq_switch(c68b9960,0,c1480b9a,269,c6883300,...) at sleepq_switch+0x15b/frame
0xdab68ac8
sleepq_wait(c6883300,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdab68af4
msleep_spin_sbt(c6883300,c6883318,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/
frame 0xdab68b2c
taskqueue_thread_loop(c68885dc,dab68ba8,0,0,0,...) at
taskqueue_thread_loop+0x15c/frame 0xdab68b64
fork_exit(c0d63900,c68885dc,dab68ba8) at fork_exit+0xa3/frame 0xdab68b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdab68b94
--- trap 0, eip = 0, esp = 0xdab68be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100023 td 0xc6855320
sched_switch(c6855320,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaaf2a58
mi_switch(104,0,daaf2ac8,c0d5f211,c6855320,c6712400) at mi_switch+0x122/frame 0xdaaf2a90
sleepq_switch(c6855320,0,c1480b9a,269,c6712410,...) at sleepq_switch+0x15b/frame 0xdaaf2ab8
sleepq_wait(c6712400,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaaf2ae4
_sleep(c6712400,c6712418,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaaf2b2c
taskqueue_thread_loop(c2025ee8,daaf2ba8,0,0,0,...) at taskqueue_thread_loop+0x11b/frame 0xdaaf2b64
fork_exit(c0d63900,c2025ee8,daaf2ba8) at fork_exit+0xa3/frame 0xdaaf2b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaf2b94
--- trap 0, eip = 0, esp = 0xdaaf2be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100021 td 0xc6855960
sched_switch(c6855960,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaaeca58
mi_switch(104,0,daaecac8,c0d5f211,c6855960,c6712680) at mi_switch+0x122/frame 0xdaaeca90
sleepq_switch(c6855960,0,c1480b9a,269,c6712690,...) at sleepq_switch+0x15b/frame 0xdaaecab8
sleepq_wait(c6712680,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaaecae4
_sleep(c6712680,c6712698,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaaecb2c
taskqueue_thread_loop(c201d81c,daaecba8,0,0,0,...) at taskqueue_thread_loop+0x11b/frame 0xdaaecb64
fork_exit(c0d63900,c201d81c,daaecba8) at fork_exit+0xa3/frame 0xdaaecb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaaecb94
--- trap 0, eip = 0, esp = 0xdaaecbe0, ebp = 0 ---

Tracing command kernel pid 0 tid 100020 td 0xc6855c80
sched_switch(c6855c80,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaae9a68
mi_switch(104,0,0,0,c6855c80,c6712700) at mi_switch+0x122/frame 0xdaae9aa0
sleepq_switch(c6855c80,0,c1480b9a,269,0,...) at sleepq_switch+0x15b/frame 0xdaae9ac8
sleepq_wait(c6712700,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaae9af4
msleep_spin_sbt(c6712700,c6712718,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/frame 0xdaae9b2c
taskqueue_thread_loop(c1ff1a74,daae9ba8,0,0,0,...) at taskqueue_thread_loop+0x15c/frame 0xdaae9b64
fork_exit(c0d63900,c1ff1a74,daae9ba8) at fork_exit+0xa3/frame 0xdaae9b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaae9b94
--- trap 0, eip = 0, esp = 0xdaae9be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100019 td 0xc6856000
sched_switch(c6856000,0,104,6681622a,0,...) at sched_switch+0x2d2/frame 0xdaae6a68
mi_switch(104,0,c6884c80,0,c6856000,c6712700) at mi_switch+0x122/frame

0xdaae6aa0
sleepq_switch(c6856000,0,c1480b9a,269,c6712700,...) at sleepq_switch+0x15b/frame 0xdaae6ac8
sleepq_wait(c6712700,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaae6af4
msleep_spin_sbt(c6712700,c6712718,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/frame 0xdaae6b2c
taskqueue_thread_loop(c1ff1a74,daae6ba8,0,0,0,...) at taskqueue_thread_loop+0x15c/frame 0xdaae6b64
fork_exit(c0d63900,c1ff1a74,daae6ba8) at fork_exit+0xa3/frame 0xdaae6b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaae6b94
--- trap 0, eip = 0, esp = 0xdaae6be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100018 td 0xc6856320
sched_switch(c6856320,0,104,20,0,...) at sched_switch+0x2d2/frame 0xdaae3a68
mi_switch(104,0,0,0,c6856320,c6712700) at mi_switch+0x122/frame 0xdaae3aa0
sleepq_switch(c6856320,0,c1480b9a,269,c6712700,...) at sleepq_switch+0x15b/frame 0xdaae3ac8
sleepq_wait(c6712700,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaae3af4
msleep_spin_sbt(c6712700,c6712718,c1467e98,0,0,...) at msleep_spin_sbt+0x1d4/frame 0xdaae3b2c
taskqueue_thread_loop(c1ff1a74,daae3ba8,0,0,0,...) at taskqueue_thread_loop+0x15c/frame 0xdaae3b64
fork_exit(c0d63900,c1ff1a74,daae3ba8) at fork_exit+0xa3/frame 0xdaae3b94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaae3b94
--- trap 0, eip = 0, esp = 0xdaae3be0, ebp = 0 ---

Tracing command kernel pid 0 tid 100015 td 0xc6856c80
sched_switch(c6856c80,0,104,c0cb43f3,0,...) at sched_switch+0x2d2/frame 0xdaadaa58
mi_switch(104,0,daadaac8,c0d5f211,c6856c80,c6712b00) at mi_switch+0x122/frame 0xdaadaa90
sleepq_switch(c6856c80,0,c1480b9a,269,c6712b10,...) at sleepq_switch+0x15b/frame 0xdaadaab8
sleepq_wait(c6712b00,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaadaae4
_sleep(c6712b00,c6712b18,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaadab2c
taskqueue_thread_loop(c2021718,daadaba8,0,0,0,...) at taskqueue_thread_loop+0x11b/frame 0xdaadab64
fork_exit(c0d63900,c2021718,daadaba8) at fork_exit+0xa3/frame 0xdaadab94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaadab94
--- trap 0, eip = 0, esp = 0xdaadabe0, ebp = 0 ---

Tracing command kernel pid 0 tid 100011 td 0xc6710640
sched_switch(c6710640,0,104,0,0,...) at sched_switch+0x2d2/frame 0xdaacda58
mi_switch(104,0,daacdac8,c0d5f211,c6710640,c6713e80) at mi_switch+0x122/frame 0xdaacda90
sleepq_switch(c6710640,0,c1480b9a,269,c6713e90,...) at sleepq_switch+0x15b/frame 0xdaacdab8

sleepq_wait(c6713e80,0,c1467e98,0,0,...) at sleepq_wait+0x3f/frame 0xdaacdae4
_sleep(c6713e80,c6713e98,0,c1467e98,0,...) at _sleep+0x2ae/frame 0xdaacdb2c
taskqueue_thread_loop(c1f94fc4,daacdba8,0,0,0,...) at taskqueue_thread_loop+0x11b/
frame 0xdaacdb64
fork_exit(c0d63900,c1f94fc4,daacdba8) at fork_exit+0xa3/frame 0xdaacdb94
fork_trampoline() at fork_trampoline+0x8/frame 0xdaacdb94
--- trap 0, eip = 0, esp = 0xdaacdbe0, ebp = 0 ---

Tracing command kernel pid 0 tid 100000 td 0xc201d360
sched_switch(c201d360,0,104,0,c201e5a4,...) at sched_switch+0x2d2/frame
0xc2422adc
mi_switch(104,0,101,c201d360,c2422b68,c201d054) at mi_switch+0x122/frame
0xc2422b14
sleepq_switch(c201d360,0,c1480b9a,28c,101,...) at sleepq_switch+0x15b/frame
0xc2422b3c
sleepq_timedwait(c201d054,54,9,0,0,...) at sleepq_timedwait+0x3f/frame 0xc2422b68
_sleep(c201d054,0,54,c14b5ec1,fffff470,...) at _sleep+0x282/frame 0xc2422bb0
swapper() at swapper+0x2c0/frame 0xc2422bf8
begin() at begin+0x22
db:0:kdb.enter.default> capture off
config.txt0600001367012754764136 7561 ustarrootwheeloptions
CONFIG_AUTOGENERATED
ident   pfSense
machine   i386
cpu   I686_CPU
cpu   I586_CPU
cpu   I486_CPU
makeoptions WITH_CTF=1
makeoptions DEBUG=-g
options   CONSPEED=115200
options   MROUTING
options   DEVICE_POLLING
options   HIFN_RNDTEST
options   HIFN_DEBUG
options   MSGTQL=2048
options   MSGSSZ=32
options   MSGSEG=512
options   MSGMNI=40
options   MSGMNB=8192
options   ALTQ_CODEL
options   ALTQ_NOPCC
options   ALTQ_FAIRQ
options   ALTQ_PRIQ
options   ALTQ_HFSC
options   ALTQ_RIO
options   ALTQ_RED
options   ALTQ_CBQ

```
options         ALTQ
options         TCP_SIGNATURE
options         IPSEC_NAT_T
options         IPSEC
options         NETGRAPH_PRED1
options         NETGRAPH_DEFLATE
options         NETGRAPH_CAR
options         NETGRAPH_PIPE
options         NETGRAPH_TCPMSS
options         NETGRAPH_TEE
options         NETGRAPH_HOLE
options         NETGRAPH_FRAME_RELAY
options         NETGRAPH_ASYNC
options         NETGRAPH_ECHO
options         NETGRAPH_CISCO
options         NETGRAPH_BRIDGE
options         NETGRAPH_ONE2MANY
options         NETGRAPH_LMI
options         NETGRAPH_KSOCKET
options         NETGRAPH_VJC
options         NETGRAPH_UI
options         NETGRAPH_MPPC_ENCRYPTION
options         NETGRAPH_TTY
options         NETGRAPH_SOCKET
options         NETGRAPH_RFC1490
options         NETGRAPH_PPTPGRE
options         NETGRAPH_PPPOE
options         NETGRAPH_PPP
options         NETGRAPH_EIFACE
options         NETGRAPH_IFACE
options         NETGRAPH_ETHER
options         NETGRAPH_BPF
options         NETGRAPH_L2TP
options         NETGRAPH_VLAN
options         NETGRAPH
options         IPSTEALTH
options         IPFIREWALL_VERBOSE
options         IPFIREWALL_DEFAULT_TO_ACCEPT
options         PPS_SYNC
options         NULLFS
options         UNIONFS
options         TMPFS
options         GEOM_BDE
options         GEOM_ELI
options         GEOM_UZIP
options         GEOM_MIRROR
options         DDB
```

```
options         XENHVM
options         HYPERV
options         ATH_ENABLE_11N
options         AH_AR5416_INTERRUPT_MITIGATION
options         AH_SUPPORT_AR5416
options         IEEE80211_SUPPORT_MESH
options         IEEE80211_AMPDU_AGE
options         SC_PIXEL_MODE
options         VESA
options         AHD_REG_PRETTY_PRINT
options         AHC_REG_PRETTY_PRINT
options         ATA_STATIC_ID
options         SMP
options         KDB
options         RCTL
options         RACCT_DEFAULT_TO_DISABLED
options         RACCT
options         INCLUDE_CONFIG_FILE
options         DDB_CTF
options         KDTRACE_HOOKS
options         PROCDESC
options         CAPABILITIES
options         CAPABILITY_MODE
options         AUDIT
options         HWPMC_HOOKS
options         KBD_INSTALL_CDEV
options         PRINTF_BUFR_SIZE=128
options         _KPOSIX_PRIORITY_SCHEDULING
options         SYSVSEM
options         SYSVMSG
options         SYSVSHM
options         STACK
options         KTRACE
options         SCSI_DELAY=5000
options         GEOM_LABEL
options         GEOM_RAID
options         GEOM_PART_GPT
options         PSEUDOFS
options         PROCFS
options         CD9660
options         MSDOSFS
options         NFS_ROOT
options         NFSLOCKD
options         NFSD
options         NFSCL
options         MD_ROOT
options         QUOTA
```

```
options         UFS_GJOURNAL
options         UFS_DIRHASH
options         UFS_ACL
options         SOFTUPDATES
options         FFS
options         SCTP
options         TCP_OFFLOAD
options         INET6
options         INET
options         PREEMPTION
options         SCHED_ULE
options         NEW_PCIB
options         NATIVE
options         GEOM_PART_MBR
options         GEOM_PART_EBR_COMPAT
options         GEOM_PART_EBR
options         GEOM_PART_BSD
options         ISAPNP
device isa
device npx
device mem
device io
device uart_ns8250
device atpic
device apic
device cpufreq
device acpi
device eisa
device pci
device fdc
device ahci
device ata
device mvs
device siis
device ahb
device ahc
device ahd
device esp
device hptiop
device isp
device mpt
device mps
device mpr
device sym
device trm
device adv
device adw
```

device aha
device aic
device bt
device ncv
device nsp
device stg
device isci
device scbus
device ch
device da
device sa
device cd
device pass
device ses
device amr
device arcmsr
device asr
device ciss
device dpt
device hptmv
device hptnr
device hptrr
device hpt27xx
device iir
device ips
device mly
device twa
device tws
device aac
device aacp
device aacraid
device ida
device mfi
device mlx
device mrsas
device pst
device twe
device nvme
device nvd
device atkbdc
device atkbd
device psm
device kbdmux
device vga
device splash
device sc
device vt

device vt_vga
device agp
device pmtimer
device cbb
device pccard
device cardbus
device uart
device ppc
device ppbus
device lpt
device ppi
device puc
device bxe
device de
device em
device igb
device ixgb
device le
device ti
device txp
device vx
device miibus
device ae
device age
device alc
device ale
device bce
device bfe
device bge
device cas
device dc
device et
device fxp
device gem
device hme
device jme
device lge
device msk
device nfe
device nge
device pcn
device re
device rl
device sf
device sge
device sis
device sk

device ste
device stge
device tl
device tx
device vge
device vr
device vte
device wb
device xl
device cs
device ed
device ex
device ep
device fe
device ie
device sn
device xe
device wlan
device wlan_wep
device wlan_ccmp
device wlan_tkip
device wlan_amrr
device an
device ath
device ath_pci
device ath_hal
device ath_rate_sample
device ipw
device iwi
device iwn
device malo
device mwl
device ral
device wi
device wpi
device loop
device random
device padlock_rng
device rdrand_rng
device ether
device vlan
device tun
device md
device gif
device faith
device firmware
device bpf

```
device uhci
device ohci
device ehci
device xhci
device usb
device ukbd
device umass
device sound
device snd_cmi
device snd_csa
device snd_emu10kx
device snd_es137x
device snd_hda
device snd_ich
device snd_via8233
device mmc
device mmcsd
device sdhci
device virtio
device virtio_pci
device vtnet
device virtio_blk
device virtio_scsi
device virtio_balloon
device hyperv
device xenpci
device vmx
device wlan_rssadapt
device wlan_xauth
device wlan_acl
device iwifw
device ipwfw
device wpifw
device iwnfw
device uath
device ralfw
device ural
device urtw
device rum
device mwlfw
device zyd
device upgt
device udav
device axe
device axge
device aue
device cue
```

device kue
device mos
device rsu
device rsufw
device run
device runfw
device rue
device urtwn
device urtwnfw
device siba_bwn
device bwn
device bwi
device pty
device ufoma
device ucom
device uslcom
device uplcom
device umct
device uvisor
device uark
device uftdi
device uvscom
device umodem
device u3g
device cdce
device uhid
device firewire
device sbp
device tap
device gre
device if_bridge
device lagg
device netmap
device enc
device pf
device pflog
device carp
device pfsync
device crypto
device cryptodev
device rndtest
device hifn
device ubsec
device safe
device padlock
device speaker
device mxge

device cxgb
device cxgbe
device nve
device oce
device gpioapu
msgbuf.txt060000174771275476413 6 7610 ustarrootwheelCopyright (c) 1992-2016 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
The Regents of the University of California. All rights reserved.
FreeBSD is a registered trademark of The FreeBSD Foundation.
FreeBSD 10.3-RELEASE-p5 #0 7307492(RELENG_2_3_2): Tue Jul 19 13:53:47 CDT 2016
root@ce23-i386-builder:/builder/pfsense-232/tmp/obj/builder/pfsense-232/tmp/FreeBSD-src/sys/pfSense i386
FreeBSD clang version 3.4.1 (tags/RELEASE_34/dot1-final 208032) 20140512
CPU: Intel(R) Core(TM) i7-3520M CPU @ 2.90GHz (2893.52-MHz 686-class CPU)
Origin="GenuineIntel" Id=0x306a9 Family=0x6 Model=0x3a Stepping=9

Features=0x783fbbf<FPU,VME,DE,PSE,TSC,MSR,MCE,CX8,APIC,SEP,MTRR,PGE,MCA,CMOV,PAT,PSE36,MMX,FXSR,SSE,SSE2>

Features2=0x5e98220b<SSE3,PCLMULQDQ,MON,SSSE3,CX16,SSE4.1,SSE4.2,POPCNT,AESNI,XSAVE,OSXSAVE,AVX,RDRAND>
AMD Features=0x8000000<RDTSCP>
AMD Features2=0x1<LAHF>
TSC: P-state invariant
real memory = 1073676288 (1023 MB)
avail memory = 1016778752 (969 MB)
pnpbios: Bad PnP BIOS data checksum
random: <Software, Yarrow> initialized
wlan: mac acl policy registered
ipw_bss: You need to read the LICENSE file in /usr/share/doc/legal/intel_ipw/.
ipw_bss: If you agree with the license, set legal.intel_ipw.license_ack=1 in /boot/loader.conf.
module_register_init: MOD_LOAD (ipw_bss_fw, 0xc081eba0, 0) error 1
ipw_ibss: You need to read the LICENSE file in /usr/share/doc/legal/intel_ipw/.
ipw_ibss: If you agree with the license, set legal.intel_ipw.license_ack=1 in /boot/loader.conf.
module_register_init: MOD_LOAD (ipw_ibss_fw, 0xc081ec50, 0) error 1
ipw_monitor: You need to read the LICENSE file in /usr/share/doc/legal/intel_ipw/.
ipw_monitor: If you agree with the license, set legal.intel_ipw.license_ack=1 in /boot/loader.conf.
module_register_init: MOD_LOAD (ipw_monitor_fw, 0xc081ed00, 0) error 1
iwi_bss: You need to read the LICENSE file in /usr/share/doc/legal/intel_iwi/.
iwi_bss: If you agree with the license, set legal.intel_iwi.license_ack=1 in /boot/loader.conf.
module_register_init: MOD_LOAD (iwi_bss_fw, 0xc08466f0, 0) error 1

iwi_ibss: You need to read the LICENSE file in /usr/share/doc/legal/intel_iwi/.
iwi_ibss: If you agree with the license, set legal.intel_iwi.license_ack=1 in /boot/loader.conf.
module_register_init: MOD_LOAD (iwi_ibss_fw, 0xc08467a0, 0) error 1
iwi_monitor: You need to read the LICENSE file in /usr/share/doc/legal/intel_iwi/.
iwi_monitor: If you agree with the license, set legal.intel_iwi.license_ack=1 in /boot/loader.conf.
module_register_init: MOD_LOAD (iwi_monitor_fw, 0xc0846850, 0) error 1
netmap: loaded module
kbd1 at kbdmux0
cryptosoft0: <software crypto> on motherboard
padlock0: No ACE support.
acpi0: <VBOX VBOXXSDT> on motherboard
acpi0: Power Button (fixed)
acpi0: Sleep Button (fixed)
attimer0: <AT timer> port 0x40-0x43,0x50-0x53 on acpi0
Timecounter "i8254" frequency 1193182 Hz quality 0
Event timer "i8254" frequency 1193182 Hz quality 100
Timecounter "ACPI-fast" frequency 3579545 Hz quality 900
acpi_timer0: <32-bit timer at 3.579545MHz> port 0x4008-0x400b on acpi0
pcib0: <ACPI Host-PCI bridge> port 0xcf8-0xcff on acpi0
pci_link2: BIOS IRQ 9 for 0.7.INTA does not match previous BIOS IRQ 10
pci0: <ACPI PCI bus> on pcib0
isab0: <PCI-ISA bridge> at device 1.0 on pci0
isa0: <ISA bus> on isab0
atapci0: <Intel PIIX4 UDMA33 controller> port 0x1f0-0x1f7,0x3f6,0x170-0x177,0x376,0xd000-0xd00f at device 1.1 on pci0
ata0: <ATA channel> at channel 0 on atapci0
ata1: <ATA channel> at channel 1 on atapci0
vgapci0: <VGA-compatible display> mem 0xe0000000-0xe07fffff irq 11 at device 2.0 on pci0
vgapci0: Boot video device
em0: <Intel(R) PRO/1000 Legacy Network Connection 1.1.0> port 0xd010-0xd017 mem 0xf0000000-0xf001ffff irq 10 at device 3.0 on pci0
em0: Ethernet address: 08:00:27:1e:49:46
em0: netmap queues/slots: TX 1/256, RX 1/256
pcm0: <Intel ICH (82801AA)> port 0xd100-0xd1ff,0xd200-0xd23f irq 11 at device 5.0 on pci0
pcm0: <SigmaTel STAC9700/83/84 AC97 Codec>
ohci0: <OHCI (generic) USB controller> mem 0xf0804000-0xf0804fff irq 11 at device 6.0 on pci0
usbus0 on ohci0
pci0: <bridge> at device 7.0 (no driver attached)
em1: <Intel(R) PRO/1000 Legacy Network Connection 1.1.0> port 0xd240-0xd247 mem 0xf0820000-0xf083ffff irq 9 at device 8.0 on pci0
em1: Ethernet address: 08:00:27:3f:1e:02
em1: netmap queues/slots: TX 1/256, RX 1/256

em2: <Intel(R) PRO/1000 Legacy Network Connection 1.1.0> port 0xd248-0xd24f mem 0xf0840000-0xf085ffff irq 11 at device 9.0 on pci0
em2: Ethernet address: 08:00:27:53:bd:6d
em2: netmap queues/slots: TX 1/256, RX 1/256
ehci0: <Intel 82801FB (ICH6) USB 2.0 controller> mem 0xf0860000-0xf0860fff irq 10 at device 11.0 on pci0
usbus1: EHCI version 1.0
usbus1 on ehci0
battery0: <ACPI Control Method Battery> on acpi0
acpi_acad0: <AC Adapter> on acpi0
atkbdc0: <Keyboard controller (i8042)> port 0x60,0x64 irq 1 on acpi0
atkbd0: <AT Keyboard> irq 1 on atkbdc0
kbd0 at atkbd0
atkbd0: [GIANT-LOCKED]
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: [GIANT-LOCKED]
psm0: model IntelliMouse Explorer, device ID 4
pmtimer0 on isa0
orm0: <ISA Option ROMs> at iomem 0xc0000-0xc7fff,0xe2000-0xe2fff pnpid ORM0000 on isa0
sc0: <System console> at flags 0x100 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
atrtc0: <AT realtime clock> at port 0x70 irq 8 on isa0
Event timer "RTC" frequency 32768 Hz quality 0
ppc0: parallel port not found.
Timecounters tick every 1.000 msec
pcm0: measured ac97 link rate at 1928 Hz
<5>em0: link state changed to UP
<5>em1: link state changed to UP
<5>em2: link state changed to UP
usbus0: 12Mbps Full Speed USB v1.0
usbus1: 480Mbps High Speed USB v2.0
ugen0.1: <Apple> at usbus0
uhub0: <Apple OHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus0
ugen1.1: <Intel> at usbus1
uhub1: <Intel EHCI root HUB, class 9/0, rev 2.00/1.00, addr 1> on usbus1
ada0 at ata0 bus 0 scbus0 target 0 lun 0
ada0: <VBOX HARDDISK 1.0> ATA-6 device
ada0: Serial Number VB9ddb6727-0736826b
ada0: 33.300MB/s transfers (UDMA2, PIO 65536bytes)
ada0: 5252MB (10757152 512 byte sectors)
ada0: Previously was known as ad0
battery0: critically low charge!
random: unblocking device.
Timecounter "TSC-low" frequency 1446761464 Hz quality 1000
Root mount waiting for: usbus1 usbus0

uhub0: 12 ports with 12 removable, self powered
Root mount waiting for: usbus1
Root mount waiting for: usbus1
Root mount waiting for: usbus1
Root mount waiting for: usbus1
uhub1: 12 ports with 12 removable, self powered
Trying to mount root from ufs:/dev/ufsid/57b1d01821cf2fd4 [rw]...
WARNING: / was not properly dismounted
<118>Configuring crash dumps...
<118>Using /dev/label/swap0 for dump device.
<118>** SU+J Recovering /dev/ufsid/57b1d01821cf2fd4
<118>** Reading 26279936 byte journal from inode 4.
<118>** Building recovery table.
<118>** Resolving unreferenced inode list.
<118>** Processing journal entries.
<118>** 66 journal records in 3584 bytes for 58.93% utilization
<118>** Freed 6 inodes (0 dirs) 2 blocks, and 5 frags.
<118>
<118>***** FILE SYSTEM MARKED CLEAN *****
<118>Filesystems are clean, continuing...
<118>Mounting filesystems...
<118>
<118>   ___
<118> ___/ f \
<118>/ p ___/ Sense
<118>___/   \
<118>   ___/
<118>
<118>Welcome to pfSense 2.3.2-RELEASE on the 'pfSense' platform...
<118>
<118>savecore: reboot after panic: ffs_valloc: dup alloc
<118>savecore: writing core to /var/crash/textdump.tar.0
<118>Creating symlinks.....ELF ldconfig path: /lib /usr/lib /usr/lib/compat /usr/local/lib /
usr/local/lib/ipsec /usr/local/lib/perl5/5.20/mach/CORE
<118>a.out ldconfig path: /usr/lib/aout /usr/lib/compat/aout
<118>done.
<118>pkg: sqlite error while executing INSERT INTO pkg_lock_pid VALUES (?1); in file
pkgdb.c:2674: UNIQUE constraint failed: pkg_lock_pid.pid
<118>pkg: Cannot get an exclusive lock on a database, it is locked by another process
mode = 0100666, inum = 17, fs = /
panic: ffs_valloc: dup alloc
cpuid = 0
KDB: enter: panic
panic.txt06000025127547641367154 ustarrootwheelffs_valloc: dup
allocversion.txt060000267127547641367637 ustarrootwheelFreeBSD 10.3-RELEASE-
p5 #0 7307492(RELENG_2_3_2): Tue Jul 19 13:53:47 CDT 2016
    root@ce23-i386-builder:/builder/pfsense-232/tmp/obj/builder/pfsense-232/tmp/

FreeBSD-src/sys/pfSense