Packet Filter (pf)

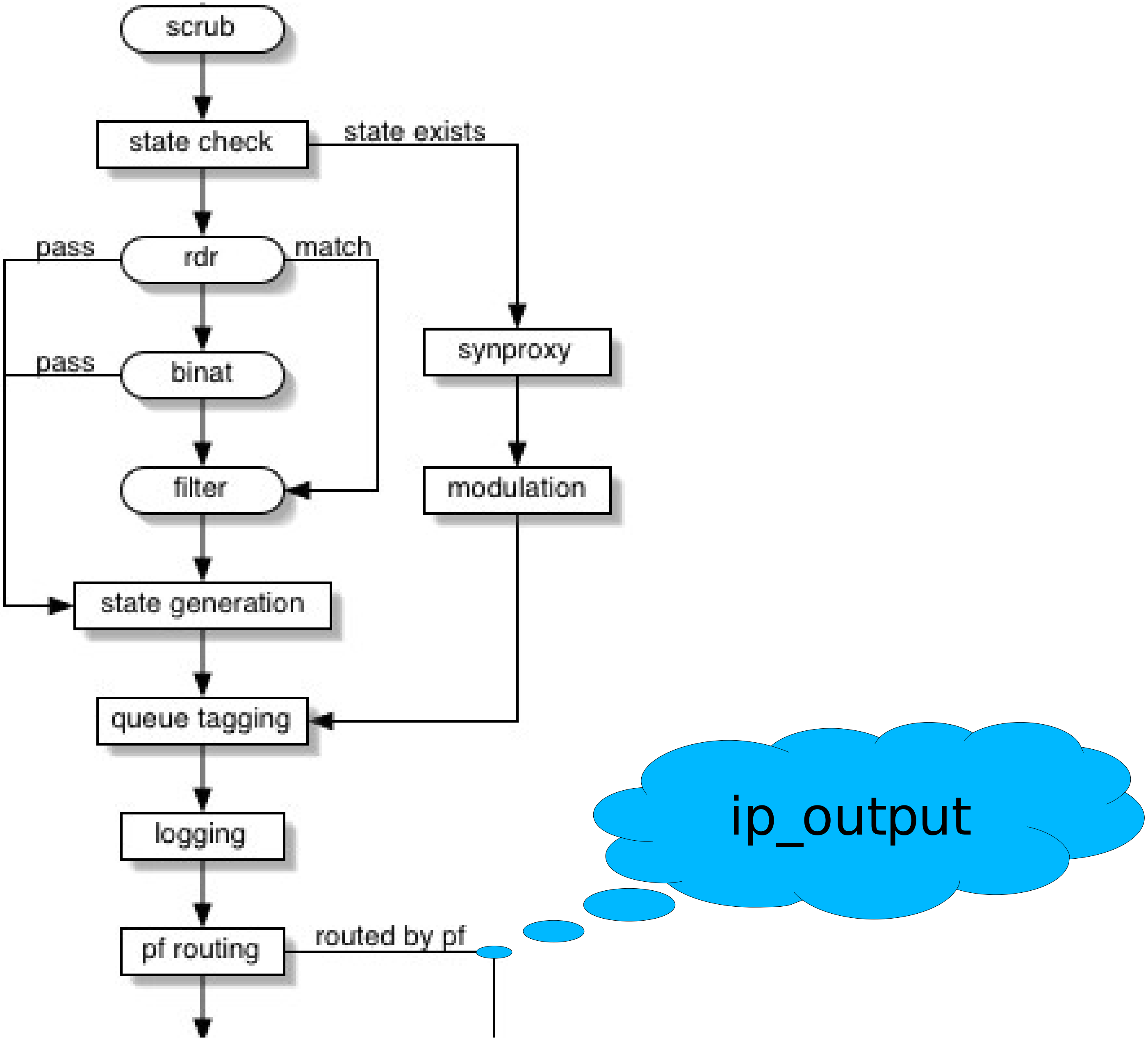
- From OpenBSD
- First imported version circa 3.4
- RELENG_5 = 3.5
- RELENG_6 = 3.7 + patches
- RELENG_7 probably 3.7 + more patches
- Beyond that – per feature imports
  - OpenBSD doesn't care for ABI/API breakage
  - FreeBSD specific features
  - Different routing code
  - *Different* SMP requirements

FreeBSD specific

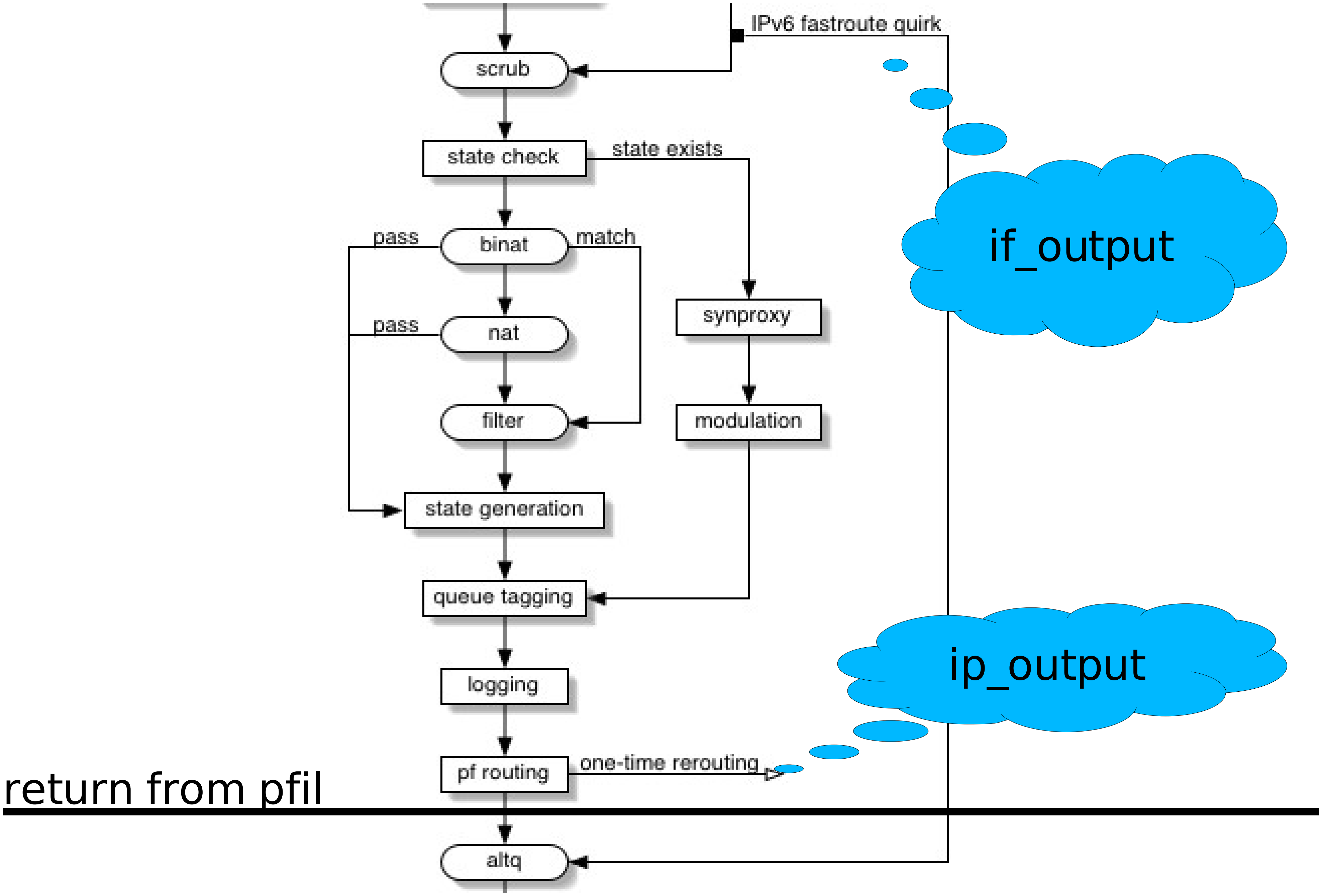
- Netgraph interaction
  - Like ipfw, but with states
  - Three SoC applicants working on it over the summer
  - Preliminary prototype available, but needs more work
- Dummynet interaction?
- Divert Sockets?
- Different more versioning friendly configuration interface (!= ioctls) ?

Packet flow

- Hooks into ip[6]_{ in,out} put via pfil(9)
  - OpenBSD has network byte order for ip_len and ip_off, hook code takes care of that
- Good reference at: http://homepage.mac.com/quension/pf/flow.png
- Basically:
  - Scrub
  - Check State
  - Process ruleset & install state
  - Post processing (log, route)


scrub
state check
state exists
pass
rdr
match
synproxy
pass
binat
filter
modulation
state generation
queue tagging
logging
pf routing
routed by pf

ip_output



if_output

ip_output

return from pfil

Concurrency opportunities

- State table and/or ruleset(s) could be accessed with rw-semantics
- Statistics gathering and state transitions still require an upgrade or per-object lock for every packet
- Difficult to get patches tested :-\

Code

pf	main checking code
_if	interface handling (name, ifnet, addr)
_ioclt	config interface, hook code, setup
_norm	scrub/normalization code
_osfp	OS fingerprinting
_subr	compat code
_table	ip table code (wrapper for radix trees)
pfvar.h	all the structures, ioctl – a bit messy :-\

if_pflog	log interface (bpf provider)
if_pfsync	sync interface (firewall failover)

Big Thank You ...

... to the **FreeBSD Foundation** for the travel grant!!!