lastline

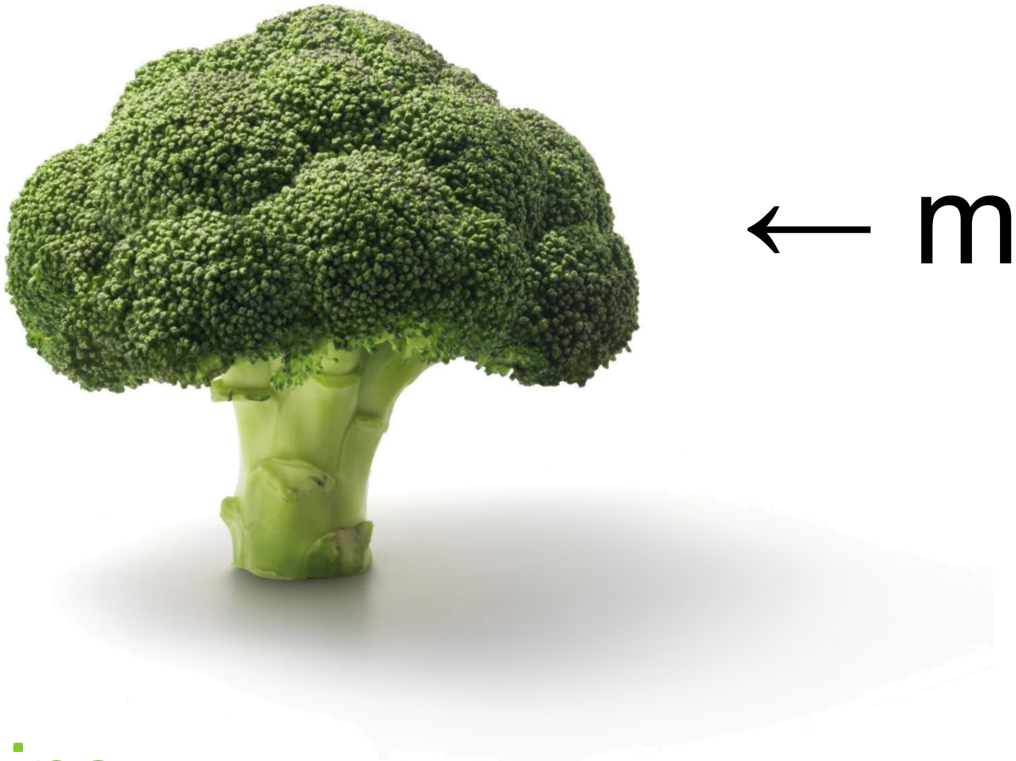
Observations on the modern NSM toolchest

Christian Kreibich
christian@lastline.com

Bro4Pros, March 2016

About me

For the Bro oldtimers

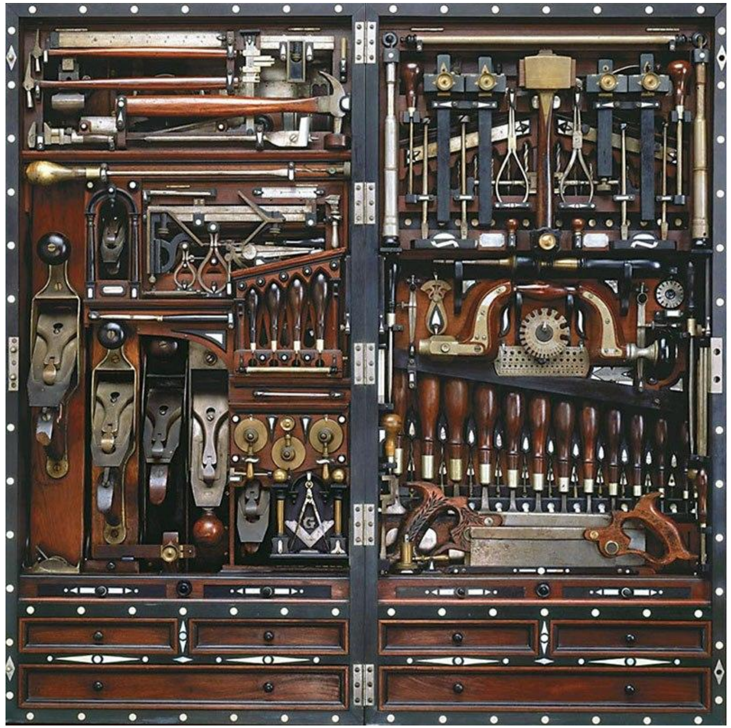


← my fault

lastline

The open-source NSM toolchest...

or

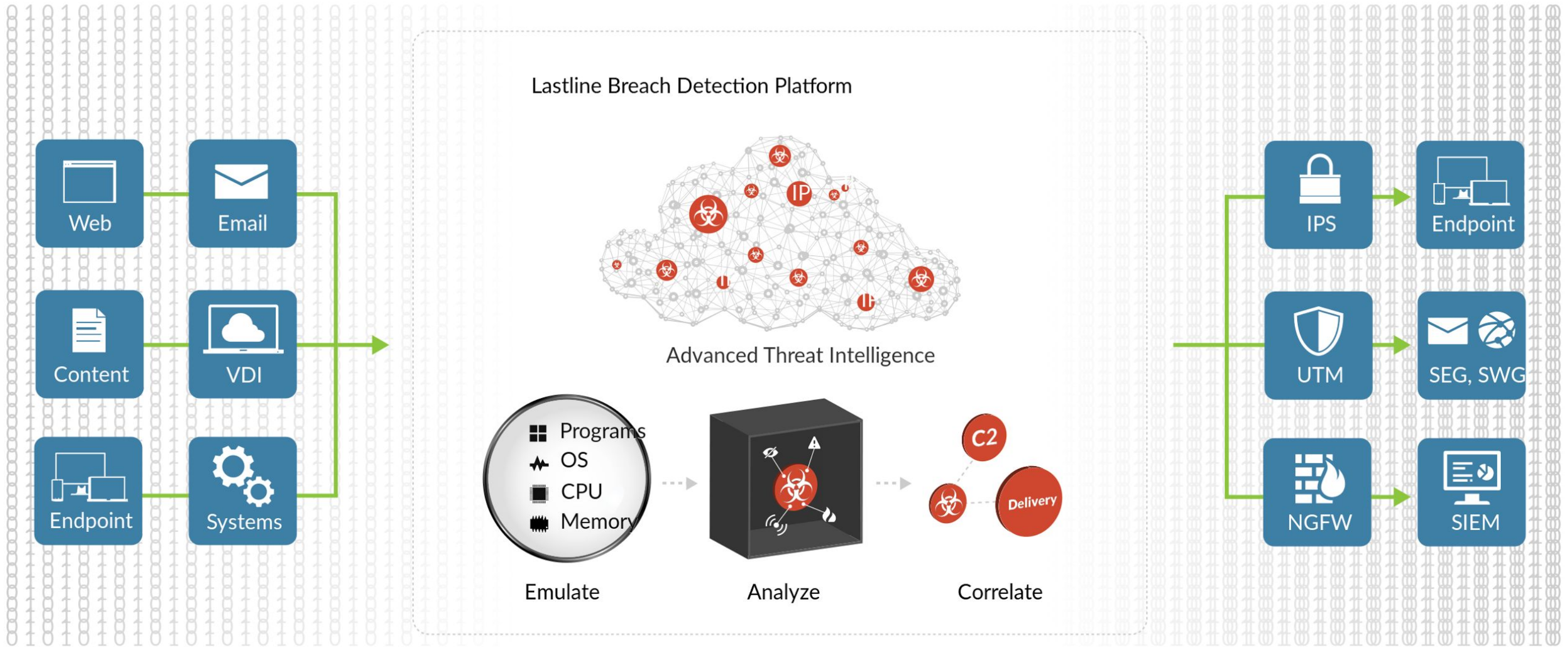


?

Background on Lastline

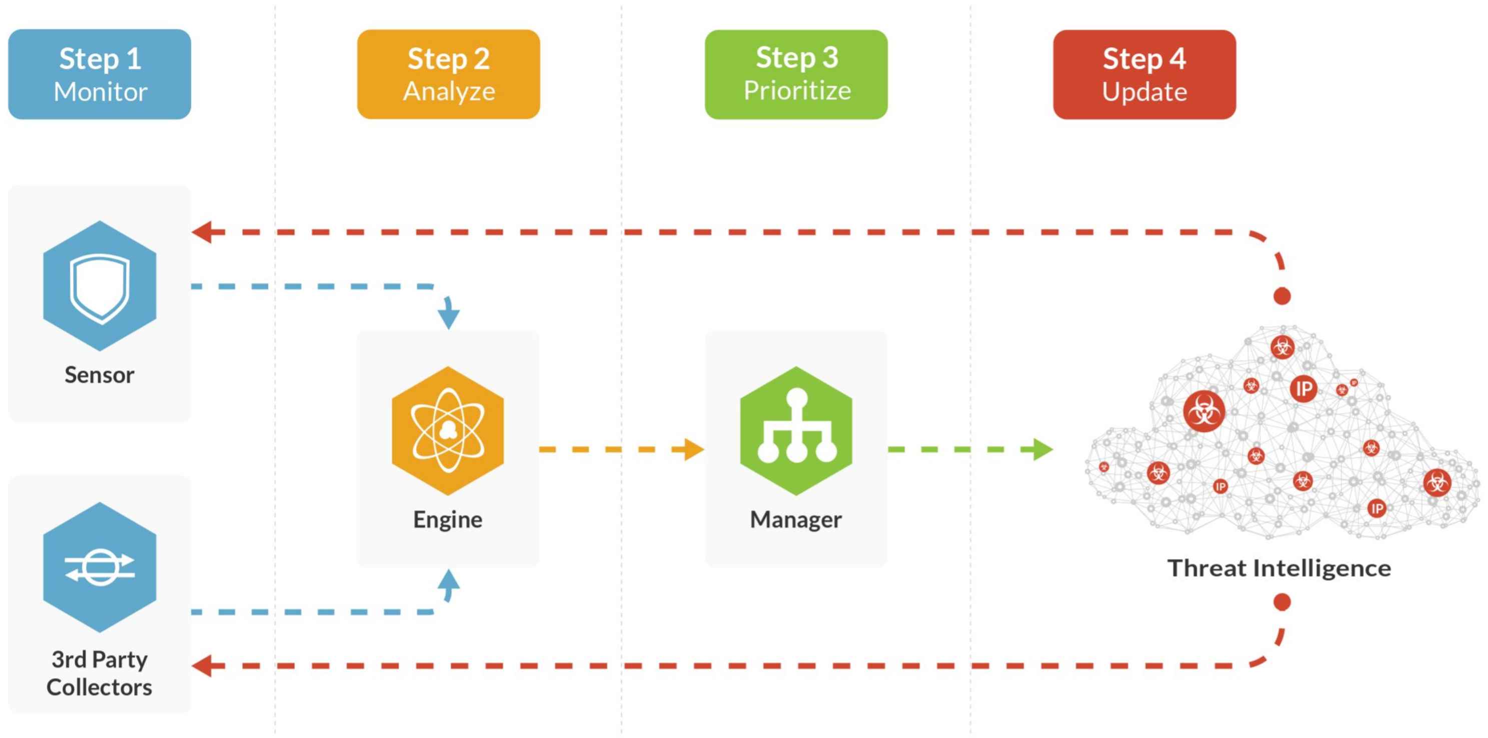
Lastline is...

- A software platform for malware protection



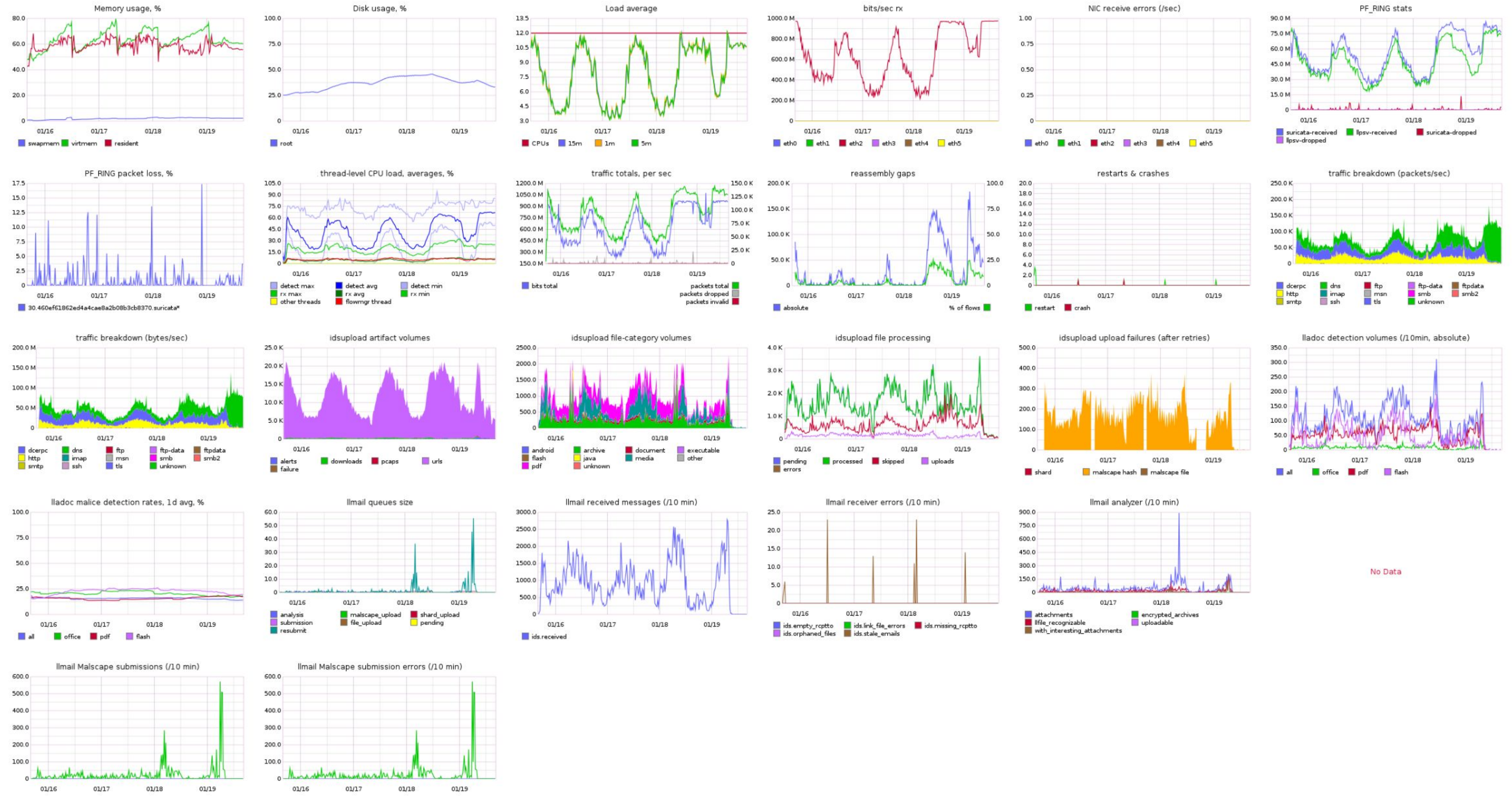
lastline

Lastline is...


Step 1
Monitor
Step 2
Analyze
Step 3
Prioritize
Step 4
Update
Sensor
Engine
Manager
Threat Intelligence
3rd Party Collectors

Linux & open-source everywhere

- Distribution based on Ubuntu packaging infrastructure, with added control
- MySQL, Cassandra, Hadoop, Ceph, RabbitMQ, ZeroMQ, Protobuf, Puppet, Ansible, Suricata, PF_RING, netmap, ...



Memory usage, %
swapmem virtmem resident
Disk usage, %
root
Load average
CPUs 15m 1m 5m
bits/sec rx
eth0 eth1 eth2 eth3 eth4 eth5
NIC receive errors (/sec)
eth0 eth1 eth2 eth3 eth4 eth5
PF_RING stats
suricata-received llpsv-received suricata-dropped llpsv-dropped
PF_RING packet loss, %
30.460ef61862ed4a4cae8a2b08b3cb8370.suricata*
thread-level CPU load, averages, %
detect max detect avg detect min rx max rx avg rx min other threads flowmgr thread
traffic totals, per sec
bits total packets total packets dropped packets invalid
reassembly gaps
absolute % of flows
restarts & crashes
restart crash
traffic breakdown (packets/sec)
dcerpc dns ftp ftp-data ftpdata http imap msn smb smb2 smtp ssh tls unknown
traffic breakdown (bytes/sec)
dcerpc dns ftp ftp-data ftpdata http imap msn smb smb2 smtp ssh tls unknown
idsupload artifact volumes
alerts downloads pcaps urls failure
idsupload file-category volumes
android archive document executable flash java media other pdf unknown
idsupload file processing
pending processed skipped uploads errors
idsupload upload failures (after retries)
shard malscape hash malscape file
lladoc detection volumes (/10min, absolute)
all office pdf flash
lladoc malice detection rates, 1d avg, %
all office pdf flash
llmail queues size
analysis malscape_upload shard_upload submission file_upload pending resubmit
llmail received messages (/10 min)
ids.received
llmail receiver errors (/10 min)
ids.empty_rcptto ids.link_file_errors ids.missing_rcptto ids.orphaned_files ids.stale_emails
llmail analyzer (/10 min)
attachments encrypted_archives llfile_recognizable uploadable with_interesting_attachments
No Data
llmail Malscape submissions (/10 min)
llmail Malscape submission errors (/10 min)
01/16 01/17 01/18 01/19

lastline

The Problem

lastline

The Lastline Sensor needs to ...

- Match industry-standard signatures
- Parse a ton of protocols
- Carve files for analysis
- Match against blacklists
- Collect basic network telemetry (NetFlow, pDNS, ...)
- Be modular & extensible
- Do a bunch of clever things I can’t talk about

The Lastline Sensor needs to ...

- Match industry-standard signatures
- Parse a ton of protocols
- Carve files for analysis
- Match against blacklists
- Collect basic network telemetry (NetFlow, pDNS, ...)
- Be modular & extensible
- Do a bunch of clever things I can’t talk about

lastline

This doesn’t exist

(as open-source)

We have tools, but no toolchest

Vortex,

...

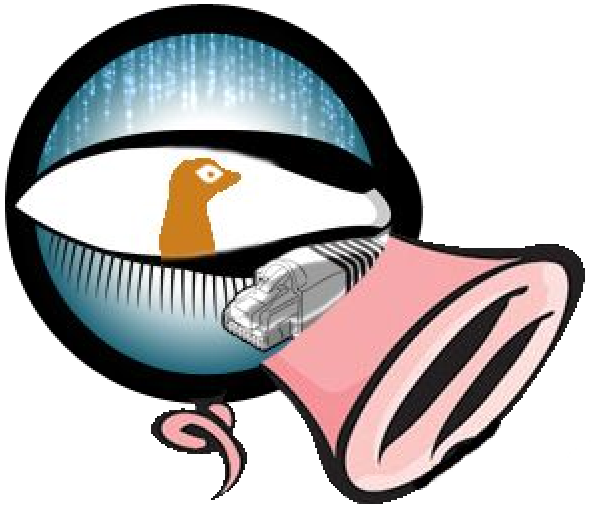
`netmap` `packetbricks` `pcap` `pf_ring`

These tools don’t mix well

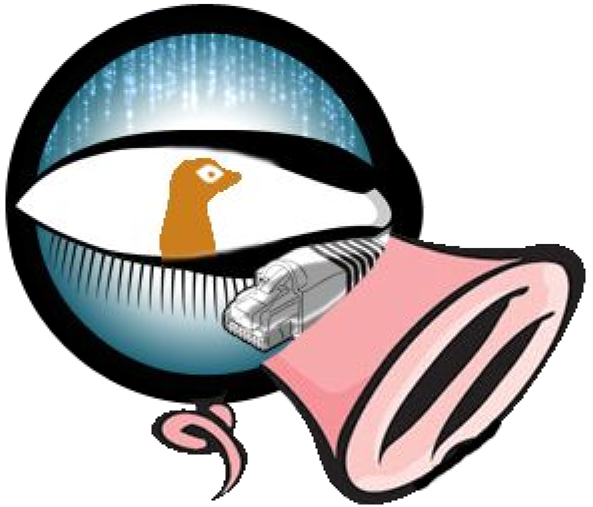
SURICATA

Vortex,

...



?



? Nope.

Wait, another Problem

We keep implementing the same stuff

Need a TCP reassembler?

libnids: dead.

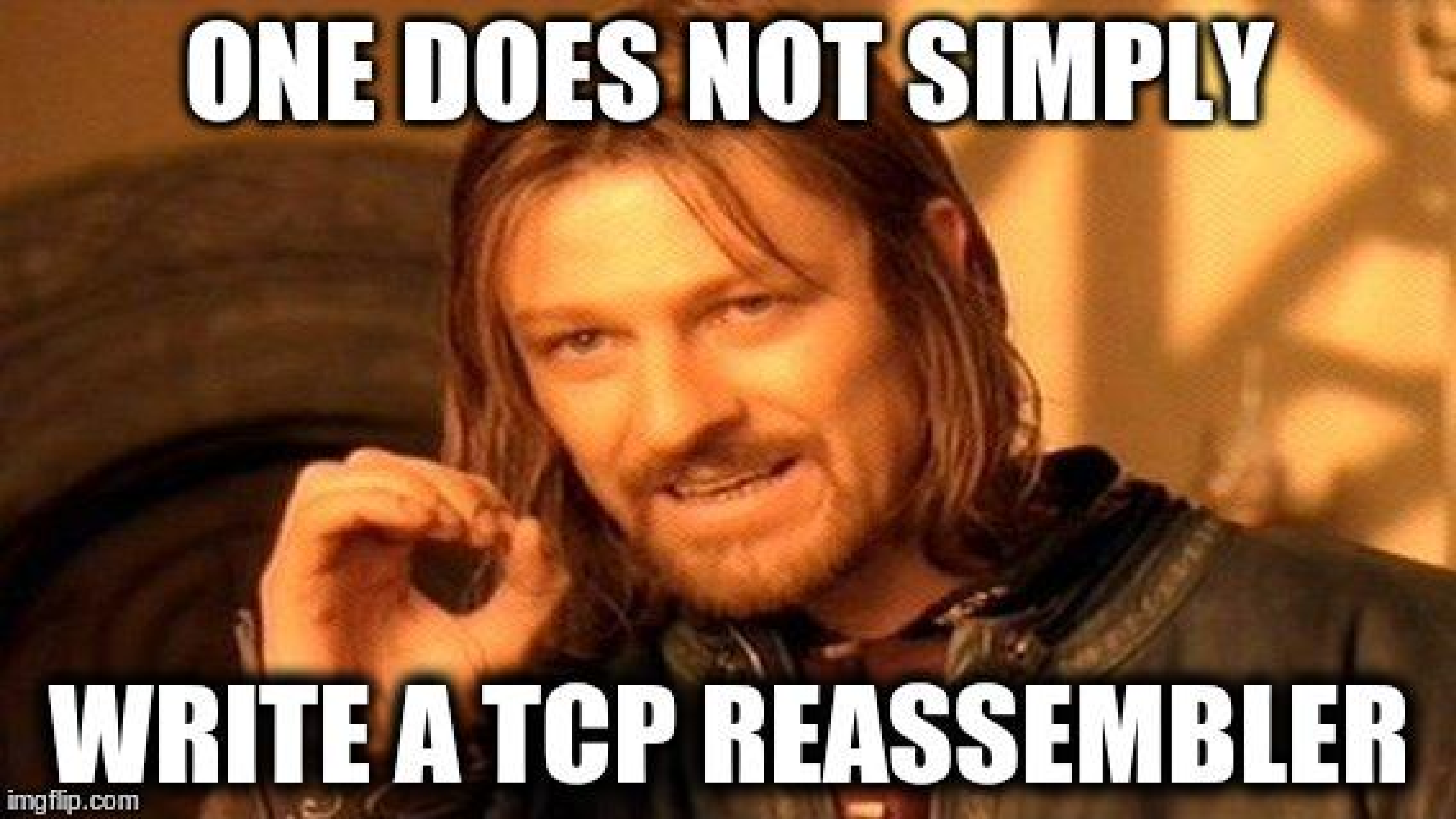
Bro: ~3,000 lines with reusable core logic

Snort: ~12,000 lines

Suricata: ~10,000 lines (excluding unit tests)

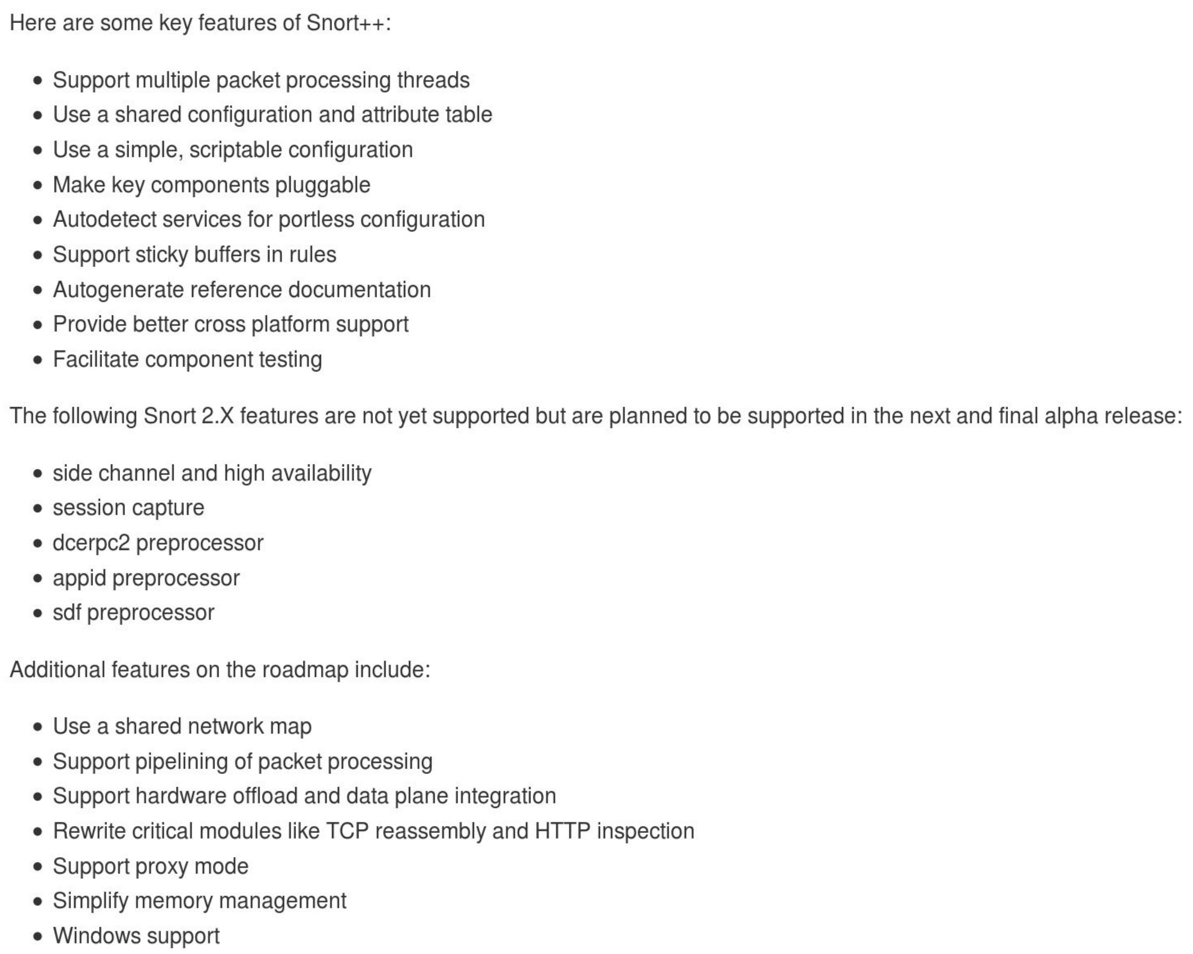
Wireshark: ~6,000 lines (excluding MPTCP)

lastline


ONE DOES NOT SIMPLY
WRITE A TCP REASSEMBLER
imgflip.com

This also applies to signature matchers and protocol parsers

It’s getting better, right?



Here are some key features of Snort++:

- Support multiple packet processing threads
- Use a shared configuration and attribute table
- Use a simple, scriptable configuration
- Make key components pluggable
- Autodetect services for portless configuration
- Support sticky buffers in rules
- Autogenerate reference documentation
- Provide better cross platform support
- Facilitate component testing

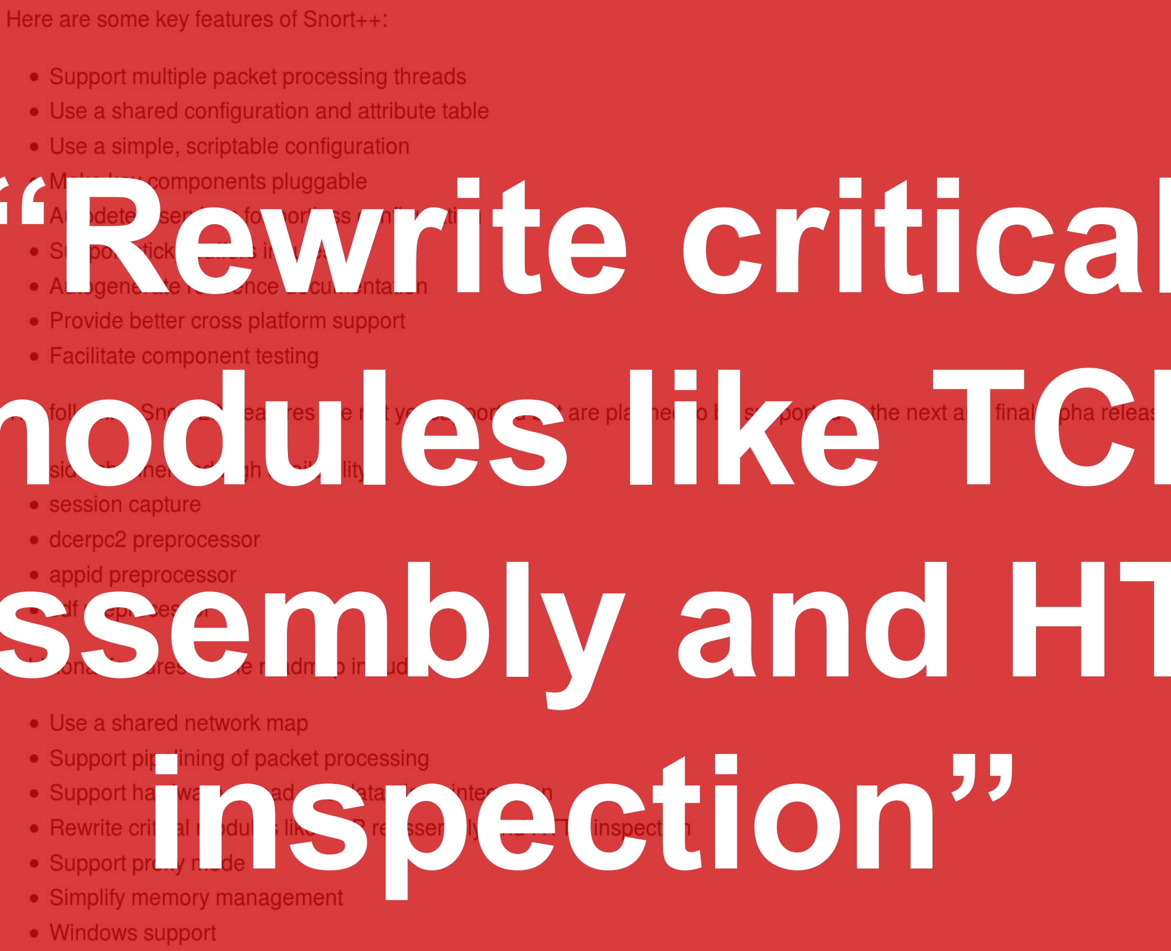
The following Snort 2.X features are not yet supported but are planned to be supported in the next and final alpha release:

- side channel and high availability
- session capture
- dcerpc2 preprocessor
- appid preprocessor
- sdf preprocessor

Additional features on the roadmap include:

- Use a shared network map
- Support pipelining of packet processing
- Support hardware offload and data plane integration
- Rewrite critical modules like TCP reassembly and HTTP inspection
- Support proxy mode
- Simplify memory management
- Windows support

“Rewrite critical modules like TCP reassembly and HTTP inspection”



Here are some key features of Snort++:

- Support multiple packet processing threads
- Use a shared configuration and attribute table
- Use a simple, scriptable configuration
- Make key components pluggable
- Autodetect services for portless configuration
- Support sticky buffers in rules
- Autogenerate reference documentation
- Provide better cross platform support
- Facilitate component testing

The following Snort features are not yet supported but are planned to be supported in the next and final alpha release:

- [illegible]
- session capture
- dcerpc2 preprocessor
- appid preprocessor
- sdf preprocessor

Additional features on the roadmap include:

- Use a shared network map
- Support pipelining of packet processing
- Support hardware offload and data plane integration
- Rewrite critical modules like TCP reassembly and HTTP inspection
- Support proxy mode
- Simplify memory management
- Windows support

lastline

Project Wishlist

libreass

- (Okay, perhaps **libtcp)**
- A community-maintained TCP stream reassembler
- Including a testsuite of quirky TCP pcaps
- With bindings for popular languages
- Could also handle IP defrag or HTTP content-range

libsigmatch

- A community-maintained signature matcher
- A de-facto community standard signature language
- Fun API challenge
- Pcap test library a plus

libprotoparse

- A community-maintained protocol parser suite

Oh wait...

http://www.icir.org/hilti/

Modular, secure,

reusable protocol parsing.

Additional Thoughts

Open-source release models matter

- Our mission is not to advance an open-source product. It is to advance our own product
- Working with a beta codebase to enjoy major fixes poses enormous risks
- Results in costly patch update rounds
- Supported stable releases increase adoption

lastline

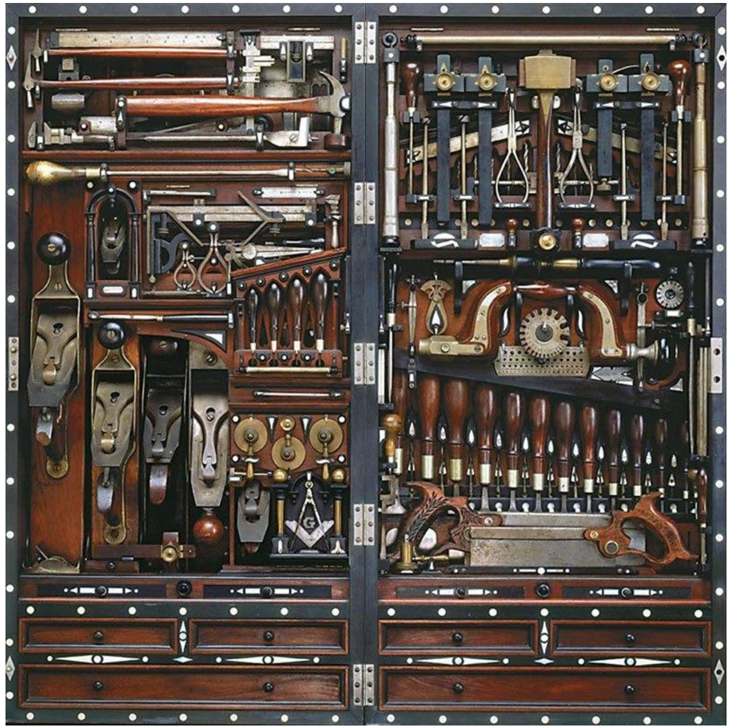
Licensing is really important

- Contagious licenses ensure open source
- Permissive licenses foster adoption
- Choose wisely!

So…

The open-source NSM toolchest...

or

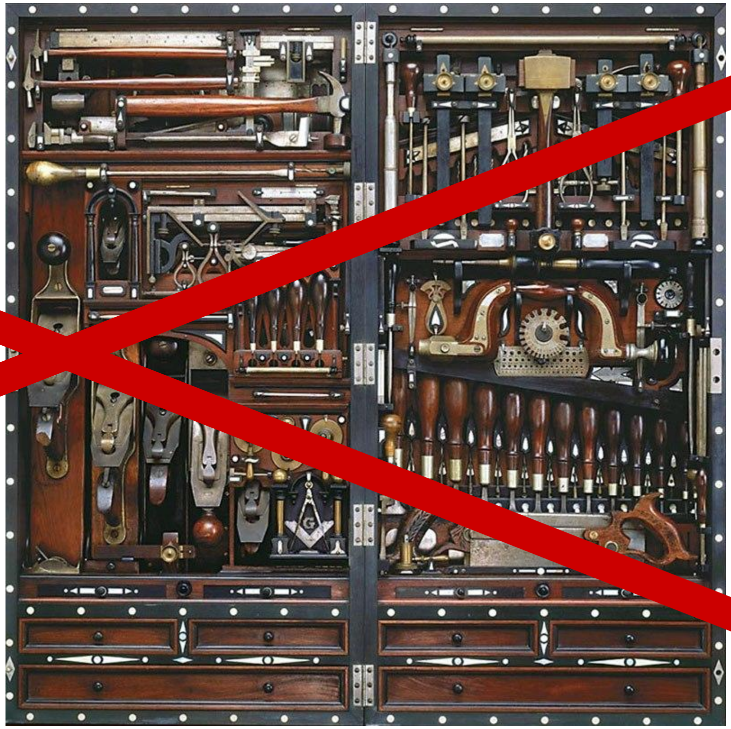


?

lastline

The open-source NSM toolchest…

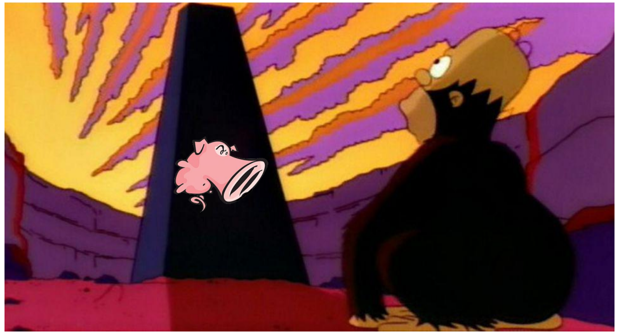
or

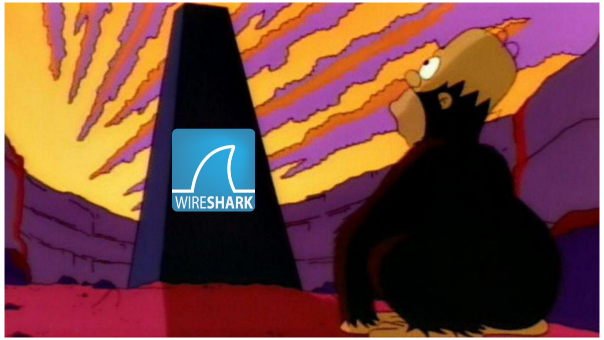


?

lastline

The open-source NSM toolchest





lastline

To be fair: these are great tools

Thanks!

(btw, Lastline is hiring)

Christian Kreibich

`christian@lastline.com`
`@ckreibich`

lastline