Observations on the modern NSM toolchest

Christian Kreibich
christian@lastline.com

Bro4Pros, March 2016
About me
For the Bro oldtimers

 ← my fault
The open-source NSM toolchest... or ?
Background on Lastline
Lastline is...

● A software platform for malware protection
Lastline is...
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, ...
The Problem
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
This doesn’t exist
(as open-source)
We have tools, but no toolchest

- netmap
- packetbricks
- pcap
- pf_ring

Vortex, ...
These tools don’t mix well
? 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)
ONE DOES NOT SIMPLY
WRITE A TCP REASSEMBLER
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
Here are some key features of Snort++:

- Support multiple packet processing threads
- Use a shared configuration and attribute table
- Use a simple, scriptable configuration
- Modules, components pluggable
- Add definitions to components
- Support rule sets
- Add agent to enhanced communication
- Provide better cross platform support
- Facilitate component testing

The following items will be added to Snort++ by the end of this month and are planned to be supported in the next alpha release:

- Support for session capture
- dcerpc2 preprocessor
- appid preprocessor
- Simplify memory management
- Windows support

“Rewrite critical modules like TCP reassembly and HTTP inspection”
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.
Licensing is really important

- Contagious licenses ensure open source
- Permissive licenses foster adoption
- Choose wisely!
So...
The open-source NSM toolchest... or ?
The open-source NSM toolchest...
The open-source NSM tool chest
To be fair: these are great tools
Thanks!
(btw, Lastline is hiring)

Christian Kreibich
christian@lastline.com
@ckreibich