SURICATA

Suricata 8 - shaping the future of network detection and prevention

OISF

Peter & Eric

Peter Manev

Co founder Stamus Networks

Suricata Evangelist, QA & Trainings at OISF

16 years Suricata contributor

Eric Leblond

Co founder Stamus Networks

Board Member at OISF

16 years Suricata contributor

Victor Julien & Peter Manev (10 years ago)

OISF

SURICATA

Network Traffic
Cloud & On-premise

SURICATA

IDS Alerts

Protocol
Transactions

Network
Flows

PCAP
Recordings

Extracted
Files

Source: Stamus Networks

OISF

SURICATA

Suricata

- Born: 2008
- Weight: 600 000 lines of code
- Composition: C, Rust
- Eat: live packets and dead ones
- Produce: JSON files/output
  - Protocol transaction
  - IDS alerts
  - PCAP
- Characteristics:
  - High speed
  - Open Source
  - Community driven
  - World famous
- Software owned and managed by the Open Information Security Foundation

OISF

SURICATA

LICENSING MODEL:

General Public License v2
Or
Proprietary license

Suricata is licensed under **GPL v2**

Ensures the software remains **free, open, and community-driven**

Guarantees freedom to run, study, modify, and share improvements

OISF

👉 FAQ on GPL: suricata.io/gpl-faqs

SURICATA

Where Suricata is developed

🇳🇱: 46.3% 🇫🇷: 16.3% 🇨🇦: 10.5% 🇮🇳: 8.5% 🇺🇸: 7.7% 🇧🇷: 2.3% 🇨🇿: 1.5% 🇳🇴: 1.0% 🇪🇸: 1.0% 🇩🇪: 1.0% 🇮🇹: 1.0% 🇮🇪: 0.5% 🇷🇺: 0.4% 🇵🇰: 0.3% 🇬🇧: 0.2% 🇨🇳: 0.2% 🇳🇿: 0.2% 🇲🇽: 0.2% 🇪🇬: 0.1% 🇵🇹: 0.1% 🇦🇺: 0.1% 🇰🇪: 0.08% 🇩🇰: 0.08% 🇨🇭: 0.08% 🇦🇹: 0.07% 🇺🇦: 0.03% 🇮🇱: 0.03% 🇹🇷: 0.03% 🇺🇬: 0.02% 🇸🇪: 0.02% 🇦🇪: 0.01% 🇫🇮: 0.01% 🇮🇷: 0.01% 🇪🇪: 0.007% 🇹🇼: 0.005% 🇮🇪: 0.003% 🇷🇴: 0.003% 🇱🇹: 0.003% 🇭🇰: 0.002%

🇪🇺: 66.7% (EU countries combined)

Non technical Suricata challenges

- A huge variety of users
  - **Mega corporations**: Google, Facebook, AWS, GM, ...
  - **Makers**: Hobbyist, Universities, Researchers, Trainers
  - Sandboxing: AnyRun, Triage, Joe's, ...
  - **Integrated appliances**: OPNsense, pfSense, ...
  - Open source IDS/NDR: Security Onion, Clear NDR Community, ...
  - **NDR vendors**: Stamus Networks, Corelight, Vectra, Cylera, Neox Networks, ...
  - Other vendors: Juniper, Proofpoint, Verizon
- Keep **community** happy and engaged
- Keep **consortium members** happy and engaged
  - Not all consortium members require a commercial license

OISF

SURICATA

Technical Suricata challenges

- High speed:
  - **100Gbps** starts to be common
  - **400Gbps** on a single server
  - But also need to run on your **raspberry**
- Work properly in real life environment
  - Input
    - Traffic capture
    - Respecting **RFC is for the weak**
    - Threat intel ingestion
  - Output
    - Export data to data lake
    - Get maximum context to the users

OISF

SURICATA

Release Cycle & Security Advisory

- Release Cycle
  - One major version every 2 years
  - Version being supported for 3 years
  - On average about one patch release every 2 months
- EOL Policy: https://suricata.io/our-story/eol-policy/
- Security Policy: https://github.com/OISF/suricata/security

How we got to Suricata 8.0

2+ years of development

- Total: **2090** files changed, **160167** insertions(+), **112455** deletions(-)
- **Rust**: 470 files changed, 62110 insertions(+), 9244 deletions(-)
- C: 1294 files changed, 70313 insertions(+), 98235 deletions(-)
- **Docs**: 158 files changed, 14237 insertions(+), 2862 deletions(-)

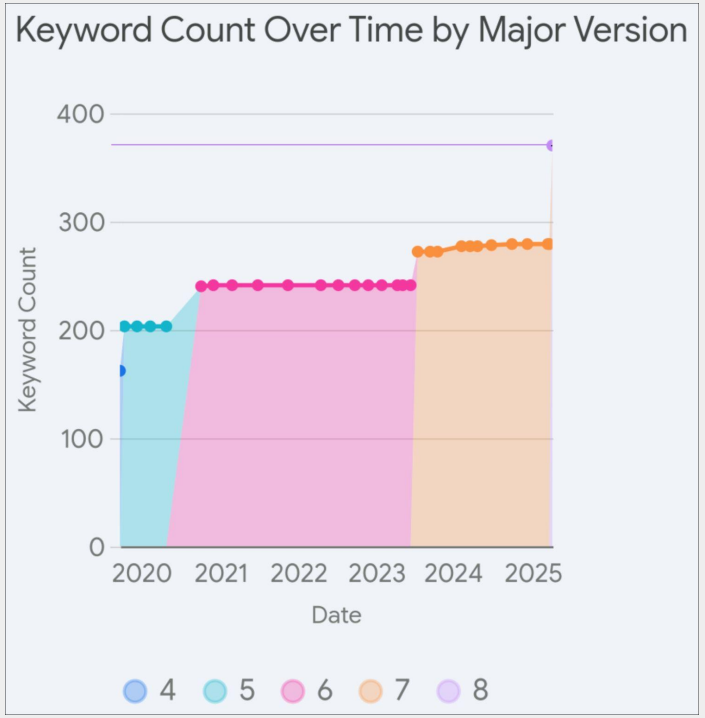
👉 Check Out: Suricata 8.0 Release Announcement

OISF

SURICATA

Improve detection capabilities

- Suricata understood (logged) protocols
  - Without offering protocol field matching capability
- Suricata 8 motto:
  - “***if you log it, you can use it in a rule***”
- Result:
  - 107 new keywords



OISF

SURICATA

Improving Detection Coverage

- Added **new** rule **keywords** for LDAP ,MIME/ EMAIL, vlan.id, DNS, SMTP, FTP, TLS, Websocket, DNS over HTTPS (DoH|DoH2)
  - To list them : ***suricata --list-keywords=csv***
- **107** new keywords added
- **38%** increase from the previous major Suricata 7 release

OISF

SURICATA

Detection limitation

- Signature language is older than some of you
- It suffers from limitation
  - At the industry level
- Directionality is one:
  - Can only match in one direction of the stream
  - Impossible to match on client direction and to server direction in one sig

Alert … **http.host**; content:"toto.com"; **http.status**; content:"200";

- Detection had to be split in 2 signatures
- And false negative could happen

Introducing transactional rules

- Express both directions in a transaction in a single rule
- Write one instead of 2 rules
  - https://docs.suricata.io/en/latest/rules/intro.html#transactional-rules

```
alert http any any => any any (file.data: to_server; content: "123";
http.stat_code; content: "500"; fast_patten;)
```

Improving IoC matching

- Dataset were introduced in Suricata 5.0
  - Match on a list of items
  - At high speed
  - Perfect for IoC ingestion
- But context is lost
  - Why is this value an IoC
    - Where does it come from ?
    - Which threat actor ?

IoC matching with dataset with JSON

- Dataset with JSON
  - Dataset but information are attached to value
  - Code contributed by Stamus Networks

Example:

http.host; dataset:isset,ioc,type string,load ioc.ndjson context_key cti, value_key ioc, **format ndjson**

In ioc.ndjson:

```
❯ cat ioc.ndjson| jq -c
{"ioc":"checkip.dyndns.org","info":"Public IP check because don't blow up yourself","context":"Legitimate but really"}
{"ioc":"ipinfo.io","info":"Public IP check because don't blow up yourself","context":"Legitimate but really an .io domain"}
{"ioc":"rlxwzlils072stb.top","info":"Cat probably walked on the keyboard","context":"Entropy is so high here"}
```

Match on “rlxwz*****.top” produces:

```
"tx_id": 4,
"alert": {
  "action": "allowed",
  "gid": 1,
  "signature_id": 1,
  "rev": 1,
  "signature": "IOC check",
  "category": "",
  "severity": 3,
  "context": {
    "cti": {
      "ioc": "rlxwzlils072stb.top",
      "info": "Cat probably walked on the keyboard",
      "context": "Entropy is so high here"
    }
  }
},
"ts_progress": "request_complete",
"tc_progress": "response_started",
"http": {
  "hostname": "rlxwzlils072stb.top",
  "url": "/installreport?r=1",
  "http_method": "GET",
  "protocol": "HTTP/1.1",
  "length": 0
},
"app_proto": "http",
"direction": "to_server",
```

Increased protocol coverage

- **Websocket** support
- **LDAP** support
- **ARP**: decoder and logger
- DNS over HTTPS (**DoH/DoH2**)
- **SIP**: parse traffic over TCP
- **SDP**: parse traffic over SIP
- **POP3**: decoder and logger
- Multicast DNS (**mDNS**)

OISF

SURICATA

Improving Suricata security

- Suricata parses all the mud of the network
  - RFC is suggestive
    - Information only
  - Attackers just love to get around
    - Getting unnoticed is one of their goal
- Two paths of improvement in Suricata 8.0
  - More Rust
    - 27% of code is now Rust
    - Brings memory safety and other benefits
  - Lua sandboxing and vendoring

Increased Rust coverage

- **Rust** coverage now includes the following additions:
  - **LibHTP**
    - Libhtp is now also built in - no dependency
  - **FTP**
  - **ENIP**
  - **MIME** parsing

OISF

SURICATA

Lua sandboxing and vendoring

- **Lua 5.4** has been “vendored” into the Suricata code base
- It **always available** by default.
- Lua scripts for detection and post stats can be deployed **regardless** of OS version support
- Run Lua in a **sandboxed** environment
- Docs: https://docs.suricata.io/en/latest/lua/libs/index.html

2038 support in Suricata

```
27 /*
28  * The SCTime_t member is broken up as
29  *  seconds: 44
30  *  useconds: 20
31  *
32  * Over 500000 years can be represented in 44 bits of seconds:
33  *  2^44/(365*24*60*60)
34  *  557855.560
35  * 1048576 microseconds can be represented in 20 bits:
36  *  2^20
37  *  1048576
38  */
39
40 typedef struct {
41     uint64_t secs : 44;
42     uint64_t usecs : 20;
43 } SCTime_t;
```

OISF

SURICATA

2 new use cases

- Feedback from the community and consortium members
  - Usage of Suricata as a firewall
    - Diverted use of IPS mode
    - Pointed by AWS and others
- Some tools wanted to benefit from Suricata capabilities
  - Use Suricata as a library

Suricata as a Library

- **Bring** your own packets and threads to Suricata
- **Dynamically** register at runtime:
  - Application protocols parsers
  - Loggers/outputs
  - Detections (keywords)
- Example code: https://github.com/OISF/suricata/blob/master/examples/lib/custom/main.c
- Completely customizable output: https://docs.suricata.io/en/latest/devguide/extending/output/index.html

Firewall is the newest running mode

Suricata can be deployed as:

- **IDS** - Intrusion Detection System (passive sniffing)
- **Firewall** - Yes, AWS uses it like that, **new mode of operation**
- **IPS** - Intrusion Prevention system (inline)
- **NSM** - Network Security Monitoring (works without rules): protocol, flow, anomaly and file transaction logging
- **FPC** - Full Pcap Capture: also possible -> Conditional PCAP Capture
- Combinations of the above like:
    - IDS + NSM + FPC
    - IDS + Conditional PCAP capture

Firewall rule example

- Run: suricata --firewall
- Rules example:

```
# deny list some hash
drop:flow tls:client_hello_done $HOME_NET any -> $EXTERNAL_NET any
(ja4.hash; content:"e7eca2baf4458d095b7f45da28c16c34"; msg:"Drop naughty
JA4"; sid:102;)
# Disallow TLS v1.0 to some destinations.
drop:flow tls:server_hello_done $HOME_NET any -> $EXTERNAL_NET any
(tls.version:1.0; msg:"TLS 1.0 not allowed"; sid:103;)
```

- Documentation: https://docs.suricata.io/en/latest/firewall/firewall-design.html

OISF

SURICATA

Conclusion

Suricata 8.0 was possible thanks to

The OISF Team

Consortium Members

PLATINUM MEMBERS

aws

proofpoint

GOLD MEMBERS

ARISTA

corelight

CYLERA

ENEA

Juniper Networks

NEOX NETWORKS

Red Piranha

verizon

OISF

SURICATA

SURICATA

Website: suricata.io

Consortium & Support: oisf.net

Community: suricata.io/community

Open Source - Open Community - Open Future

OISF

Community Contributors

Organizations

- **stamus-networks.com**: 97 commits, code +4675 -2143. Tickets 13. Score 4810
- **cyber.gc.ca**: 4 commits, code +24261 -1467. Tickets 2. Score 2920
- **corelight.com**: 15 commits, code +529 -76. Tickets 7. Score 1305
- **endace.com**: 3 commits, code +16 -23. Tickets 4. Score 523
- **rapid7.com**: 6 commits, code +1155 -722. Tickets 1. Score 521
- **alliedtelesis.co.nz**: 3 commits, code +81 -15. Tickets 2. Score 325
- **broadcom.com**: 2 commits, code +698 -20. Tickets 1. Score 241
- **ntop.org**: 1 commits, code +707 -0. Tickets 1. Score 203
- **napatech.com**: 2 commits, code +4 -75. Tickets 0. Score 59

OISF

Source: https://inliniac.net/leaderboard/8/

SURICATA

And *Individual* Contributors

Top 25 individuals

1. 🥇 **Eric Leblond** 🥇 : 96 commits, code +4664 -2136. Tickets 11. Score 4565
2. 🥈 **Giuseppe Longo** 🥈 : 56 commits, code +6941 -1179. Tickets 11. Score 3783
3. 🥉 **Alice Akaki** 🥉 : 41 commits, code +3405 -773. Tickets 19. Score 3708
4. **Todd Mortimer**: 1 commits, code +24014 -1384. Tickets 1. Score 2689
5. **jason taylor**: 53 commits, code +2164 -1503. Tickets 4. Score 2461
6. **Sascha Steinbiss**: 14 commits, code +2041 -377. Tickets 6. Score 1256
7. **Jeff Lucovsky**: 11 commits, code +327 -58. Tickets 6. Score 1040
8. **Nathan Scrivens**: 5 commits, code +2719 -16. Tickets 3. Score 748
9. **Nancy Enos**: 5 commits, code +400 -1087. Tickets 4. Score 711
10. **Adam Kiripolsky**: 6 commits, code +610 -176. Tickets 3. Score 644

Source: https://inliniac.net/leaderboard/8/

More Resources

- Read the **Docs**: https://docs.suricata.io/en/latest/
- More Suricata **trainings/webinars**: https://suricata.io/learn/
- **Awesome Suricata** links: https://github.com/satta/awesome-suricata
- Suricata 8 release **blogs**:
  - https://forum.suricata.io/t/suricata-8-0-0-released/5854
  - https://suricata.io/2025/07/23/suricata-8-always-evolving-constantly-improving/

OISF

SURICATA

Get Involved

Community Forum
forum.suricata.io

Online Chat (Discord)
discord.gg/t3rV2x7MrG

OISF

SURICATA

Our Community

GitHub

Discourse

Discord

Suricata Newsletter

Suricata thrives because of its people—developers, researchers, and defenders worldwide who bring expertise, code, and ideas to keep the project strong.

Collaboration happens every day in our community channels and each year at SuriCon, where we come together to share knowledge, celebrate progress, and shape the future of open-source security.

OISF

👉 Community Channels: suricata.io/join-our-community/

SURICATA

Get Involved

- **Contribute** – Code, test, or improve documentation
- **Engage** – Join discussions, ask questions, and share knowledge
- **Support** – Attend trainings, sponsor, or become a Consortium Member

👉 Join Us: suricata.io/community

OISF

SURICATA

IDEAS AND BRAINSTORMING

DO NOT USE