Community ID

Standardized flow hashing for your NSM tools

Christian Kreibich
christian@corelight.com
@ckreibich

The problem

Typical Suricata log entries (eve.json)

```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    "category": "Misc activity",
    ...
}
```

```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "event_type": "http",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "tx_id": 0,
  "http": {
    "http_port": 0,
    "url": "/scripts/..%c1%9c../...",
    "http_method": "GET",
    "length": 0
  }
}
```

Typical Suricata log entries (eve.json)

```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    "category": "Misc activity",
    ...
}
```

```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "event_type": "http",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "tx_id": 0,
  "http": {
    "http_port": 0,
    "url": "/scripts/..%c1%9c../...",
    "http_method": "GET",
    "length": 0
  }
}
```

Typical Suricata log entries (eve.json)


```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    "category": "Misc activity",
    ...
}
```



```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "event_type": "http",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "tx_id": 0,
  "http": {
    "http_port": 0,
    "url": "/scripts/..%c1%9c../...",
    "http_method": "GET",
    "length": 0
  }
}
```

Typical Zeek log entries

conn.log


```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "proto":"tcp",
  "service":"http",
  "duration":6.582984,
  ...
}
```


http.log


```
{
  "ts":1052146263.269431,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "trans_depth":1,
  "method":"GET",
  "uri": ...,
  "request_body_len": 0,
  ...
}
```

Typical Zeek log entries

conn.log


```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "proto":"tcp",
  "service":"http",
  "duration":6.582984,
  ...
}
```


http.log


```
{
  "ts":1052146263.269431,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "trans_depth":1,
  "method":"GET",
  "uri": ...,
  "request_body_len": 0,
  ...
}
```

Typical Zeek log entries

conn.log

```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.[illegible]
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "proto":"tcp",
  "service":"http",
  "duration":6.582984,
  ...
}
```

http.log

```
{
  "ts":1052146263.269431,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "trans_depth":1,
  "method":"GET",
  "uri": ...,
  "request_body_len": 0,
  ...
}
```

Typical Zeek-and-Suricata log entries

```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "proto":"tcp",
  "service":"http",
  "duration":6.582984,
  ...
}
```

```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    "category": "Misc activity",
    ...
}
```

Typical Zeek-and-Suricata log entries

```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "proto":"tcp",
  "service":"http",
  "duration":6.582984,
  ...
}
```

```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    "category": "Misc activity",
    ...
}
```

Typical Zeek-and-Suricata log entries

```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "proto":"tcp",
  "service":"http",
  "duration":6.582984,
  ...
}
```

```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    "category": "Misc activity",
    ...
}
```

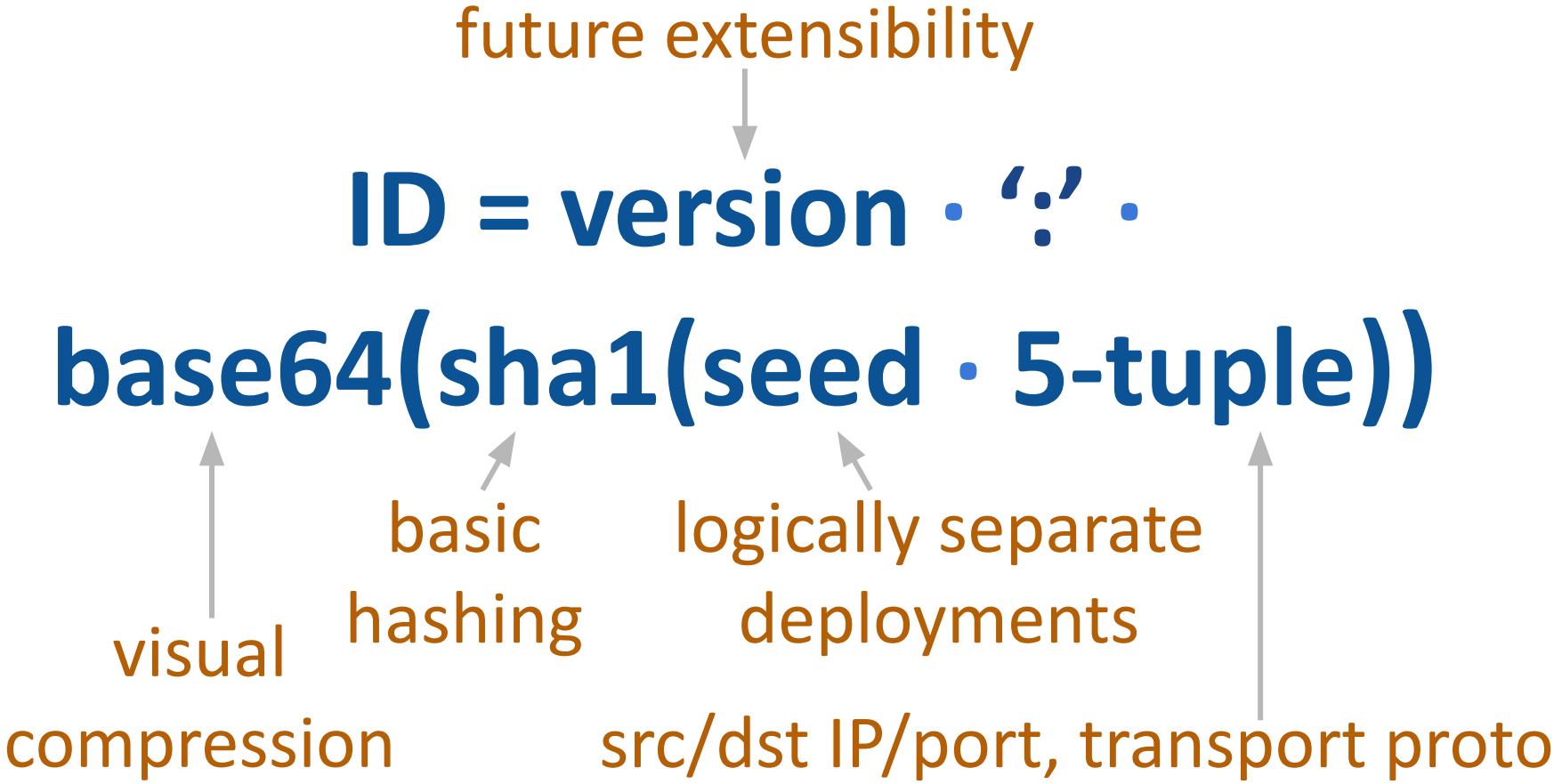
Typical Zeek-and-Suricata log entries

```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "proto":"tcp",
  "service":"http",
  "duration":6.582984,
  ...
}
```


```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    "category": "Misc activity",
    ...
}
```

Community ID

Standardized flow hashing for your NSM tools


future extensibility
ID = version · ':' ·
base64(sha1(seed · 5-tuple))
visual compression
basic hashing
logically separate deployments
src/dst IP/port, transport proto

Typical Zeek-and-Suricata log entries

```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "proto":"tcp",
  "service":"http",
  "duration":6.582984,
  ...
}
```

```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    ...,
}
```

Typical Zeek-and-Suricata log entries

```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"80.4.124.41",
  "id.resp_p":80,
  "proto":"tcp",
  "service":"http",
  "duration":6.582984,
  ...,
  "community_id":
    "1:ZEYOYMeyZNQC9DAdgsBZCtiTKqw="
}
```

```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  "src_ip": "203.241.248.20",
  "src_port": 3051,
  "dest_ip": "80.4.124.41",
  "dest_port": 80,
  "proto": "TCP",
  "community_id":
     "1:ZEYOYMeyZNQC9DAdgsBZCtiTKqw=",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    ...,
```

Typical Zeek-and-Suricata log entries


```
{
  "ts":1052146262.937361,
  "uid":"CVKjZo2GrV8DM0Fvo5",
  "id.orig_h":"203.241.248.20",
  "id.orig_p":3051,
  "id.resp_h":"[illegible]",
  "id.resp_p":8[illegible],
  "proto":"tcp",
  "service":"ht[illegible]",
  "duration":6.582984,
  ...,
  "community_id":
    "1:ZEYOYMeyZNQC9DAdgsBZCtiTKqw="
}
```


DOUBLE WOOT!


```
{
  "timestamp": "2003-05-05T07:51...",
  "flow_id": 23963675020689,
  "pcap_cnt": 10,
  "event_type": "alert",
  [illegible]
  [illegible]
  "dest_port": 80,
  "proto": "TCP",
  "community_id":
     "1:ZEYOYMeyZNQC9DAdgsBZCtiTKqw=",
  "alert": {
    "action": "allowed",
    "gid": 1,
    "signature_id": 9999999,
    "rev": 1,
    "signature": "PWNED!",
    ...,
```

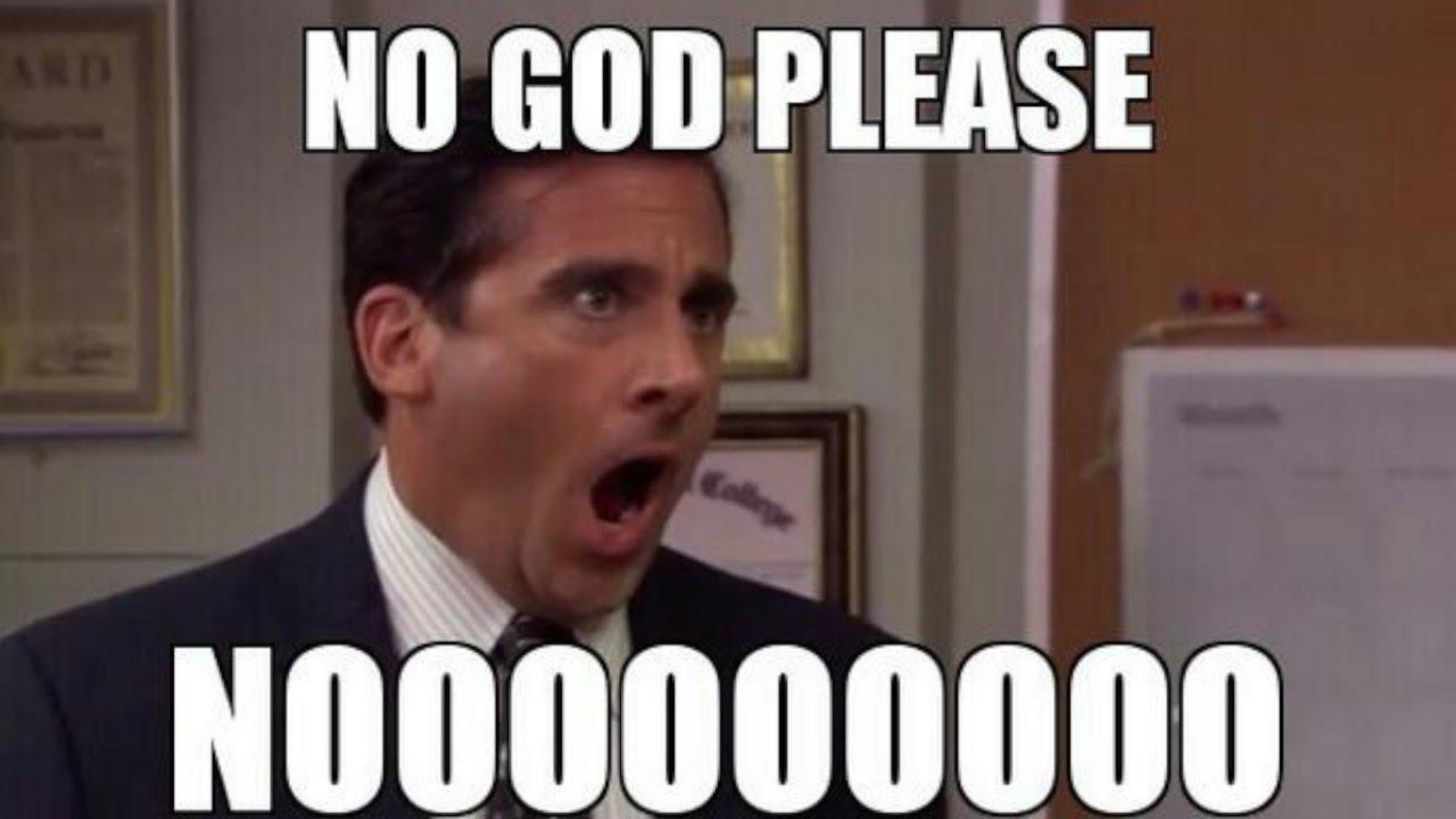
Example use cases

- Flow correlation across multiple log streams
  - Suricata & Zeek, SIEM correlation, etc
  - Also multiple installations of same system
- Associating unidirectional flows in asymmetric routes
  - Via the fine folks at ESnet — thx!
- Standardized data enrichment in SIEM

Current status

- Spec, reference implementation & data, third-party support:
  https://github.com/corelight/community-id-spec
- Implementations in C, Golang, Java, Python, Ruby
- In Suricata since 4.1:
  https://github.com/victorjulien/suricata/tree/feature/flow-community-id/v17
- In Wireshark since 3.4:
  https://corelight.blog/2020/10/07/community-id-support-for-wireshark/
- Zeek package for 2.5+:
  https://github.com/corelight/zeek-community-id

Current status


NO GOD PLEASE
NOOOOOOOOO

Yes, this isn’t feature-complete

- “v1” explicitly targets ease of implementation
  - Get people to run it and provide feedback
- Possible version-2 addition: configuration
  - Example: include VLAN, Q-in-Q, others (“**v**lan”)
  - Example: other hashing algorithm (“sha25**6**”)
  - Example: no base encoding (“**n**obase”)

    `2:v6n:f34de81f113ae0bd97242a18d1b82ddea1ef9fd4`

Performance vs. collisions

- Again, SHA-1 was our choice because of easy availability
- There are many other (non-secure) hash functions
  - Murmur2, djb2, …
- No complaints about speed or collisions so far

Factoring in time

- Currently not included
- Obvious use of time windows induces risk of ID divergence
  - Example: rounding to nearest day means a change just before and after midnight — multiple monitors may yield different results
- Clever ideas for “time-approximate fuzzy hashing” welcome

Other base encodings

- Zeek uses base62 for its ID strings

  `Ea6PGGTh0j80lGYQNskx1l3Az6C`

- Bitcoin uses base58

  `2Raiz2fPQxk3E4hERp1zStCMaN8b`

- Length vs. parse-/readability vs. performance tradeoff

Anonymity vs. reversibility

- Hashing means the ID is not reversible
- This enables certain use cases
  - “Have you seen this flow?”
- But reversibility may be desirable
  - You could simply write it out …

    `127.0.0.1/80/192.168.12.234/3434/tcp`

Thanks!

- Check out the spec, current implementations, and open issues on GitHub:

  ➜ https://github.com/corelight/community-id-spec

- If you're adding a new language implementation or product support, let us know!