



Zeek 9 and Beyond

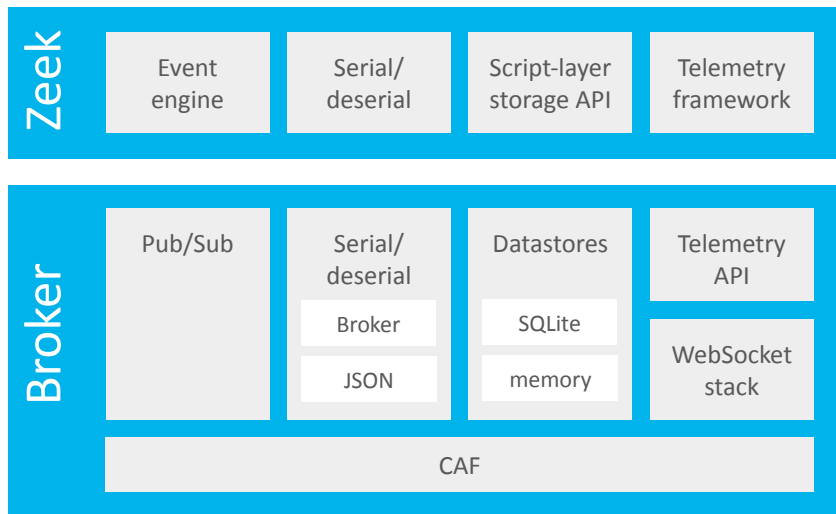
A Development Update

Christian Kreibich
christian@zeek.org

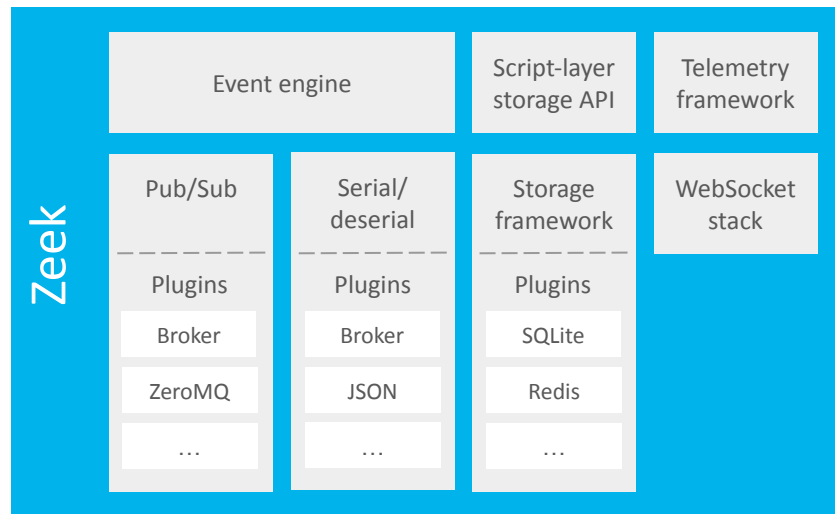
Zeek 8 to 9

Zeek 6 -> Zeek 9 architectural revamp

Zeek 6

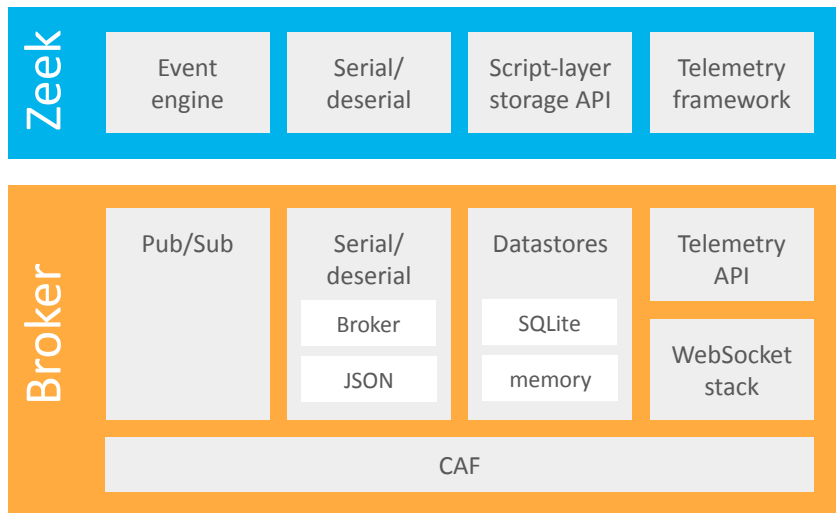


Zeek 9

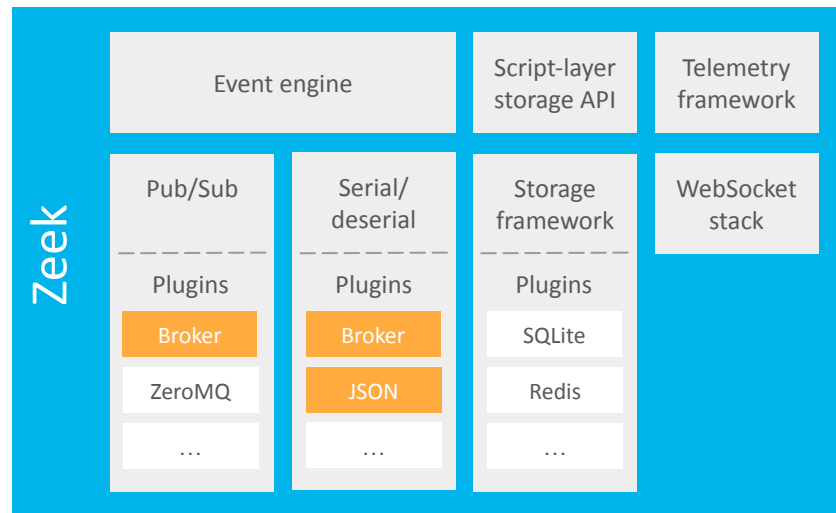


Zeek 6 -> Zeek 9 architectural revamp

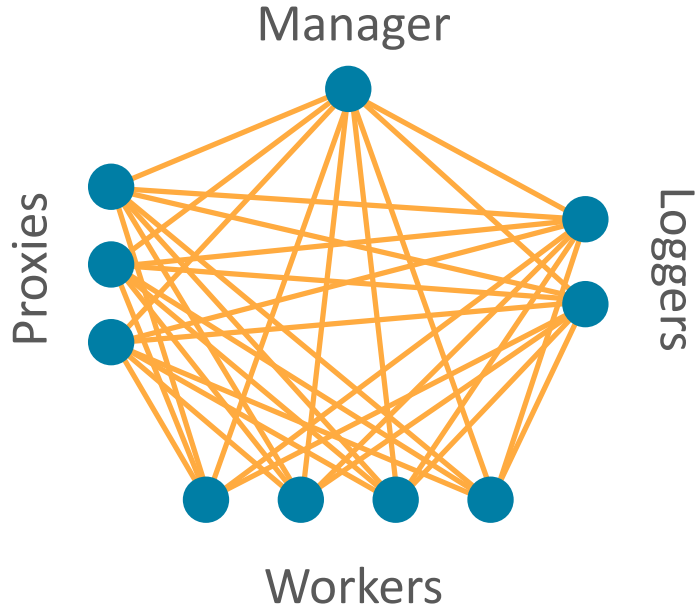
Zeek 6



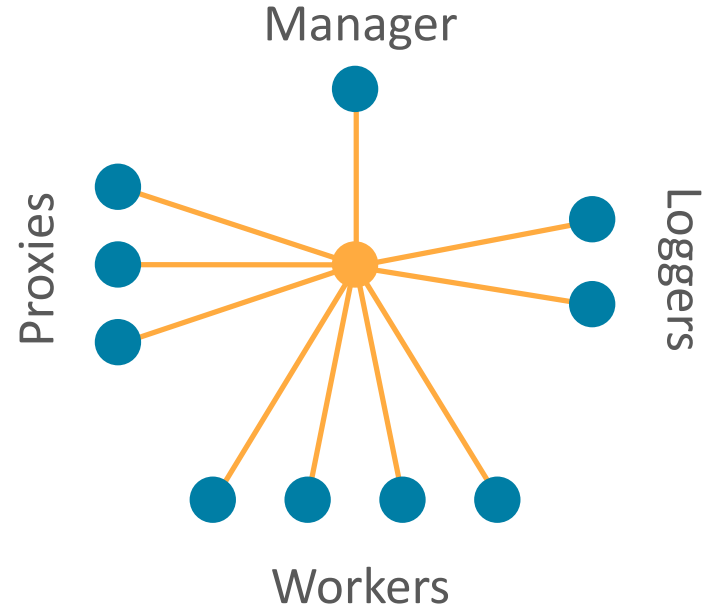
Zeek 9



Broker



ZeroMQ



- ZeroMQ available since 8.0, off by default
- ZeroMQ on by default in Zeek 8.1, easily reverted to Broker
- Zeek 8.2 adds ZeroMQ encryption support

A new state propagation mechanism

```
table state[string] of count &publish_on_change=[  
    $changes=set(TABLE_ELEMENT_NEW),  
    ...  
];
```

- New in 8.2, to replace memory-only Broker-backed tables & datastores
- A more explicit, non-magic replacement of **&synchronized**
- Use the Storage Framework for persistence & centralization
- Explicit eventing remains a viable — and now easier — option

Zeek 9: a simpler, modern cluster

- ZeroMQ as a mature, fast, true pub/sub cluster backend
- A new WebSocket stack and client libraries for talking to Zeek
 - Focus on event pub/sub, not low-level I/O, in your apps
 - Python, C, Rust, and JavaScript provided, more by community members
- JavaScript for talking from Zeek to external APIs
- New UDP packet source natively processes encapsulated traffic
 - “Zeek as a server”, with scalability via SO_REUSEPORT
 - No sniffing-based workarounds needed for cloud environments
- systemd-native cluster management as a zeekctl alternative
 - Zeek now ships with a unit file generator
 - Cleanly separates process orchestration from Zeek script configuration
 - zeekctl remains the classic alternative
- Lots more: analyzer improvements, language features, Spicy bumps, etc

Beyond Zeek 9

Zeek 9 should ship on 9/14.

Forward-looking statements follow. 🙌

Building on extensible flow tuples

- Zeek 8 introduced pluggable “conn_keys” — the underpinnings of flow tuples (e.g. to include VLANs, Geneve/VXLAN VNIs)
- This was half of the story: endpoints vs addresses is the other

```
type Endpoint: record {  
    ctx: conn_id_ctx;  
    h: addr;  
};
```

- Resembles the `conn_id` expansion in Zeek 8
- One-time `addr` -> `Endpoint` code refactoring
- Related: automatic scoping of flows to tunnels

Analyzer improvements

- Zeek currently doesn't support directional protocol matching
- Can cause confusing logs, e.g. in protocol mismatches during site scans
 - When an HTTP client contacts an SSH server, what protocols are active? Both? Neither?
 - Nothing in conn.log services, http.log has request, analyzer.log has violation, no ssh.log
- Needs directional confirmations, directional services, edit pass over analyzers
- Likely an opt-in to enrich conn.log, for initial version
- The File Analysis framework is ripe for improvements too
 - Align analyzer confirmation/violation handling with protocols
 - Improve file rotation/cleanup

A new zkg and package/plugin management

- zkg has not seen updates in years but there's lots to do:
 - git shines through (or masks errors)
 - It's too hard to develop packages with it
 - Bundles have strange failure modes
 - It never cleans up after itself
 - Most don't want testing at installation time
 - Plugin-packages have very subtle failure modes
- We will likely stop bundling zkg with Zeek
 - ... but keep it working seamlessly with your local Zeek installation(s)
 - Streamlines Python-native dependency management
- Expect frequent release previews over the next months

The last bits of the Broker offramp

- **./configure --enable-broker**
 - Broker-free builds are coming 🌟
 - Currently every Zeek still has a broker_mgr
- Alternatives to Broker's built-in event/log data serializers
 - We're relying on Broker for JSON and its own binary format
 - Nice opportunity to consider off-the-shelf formats
 - We already liked Cap'n Proto before we embarked on ZeroMQ
- Several frameworks still rely on Broker, too
 - Control, NetControl, Known, Supervisor

A few things are leaving

- The Management Framework is deprecated for removal in 9.1
 - Hard-to-maintain code due to event-chained “RPCs”
 - Concepts like node-local commands will remain, in a revised Control framework
- All of Broker is deprecated at this point, for removal in 10.1
- We had to sunset FreeBSD support
 - Mainly due to lack of reasonable CI support options
 - Errors unique to FreeBSD were really quite rare
 - And tended to be community-reported anyway :-)

And in the meantime...

- A TON of security issues thanks to multiple efforts:
 - OpenAI & Trail of Bits
 - Anthropic & ADA Logics
 - The Canadian Communications Security Establishment
 - Our own scanning, thanks to Corelight
- 8.0.10 and 8.2.1 had over two dozen fixes, majority HIGH
- We recently closed ~50, and are currently about 40% through 70 more
- This is awesome 🌟 but can't be all we do
- We've changed several aspects of our security policies as a consequence of this — more details soon

Thanks!

Documentation

<https://docs.zeeq.org>

Community links

<https://zeeq.org/community>

Github project

<https://github.com/zeeq>

Project wiki

<https://github.com/zeeq/zeeq/wiki>

Zeek is



!

