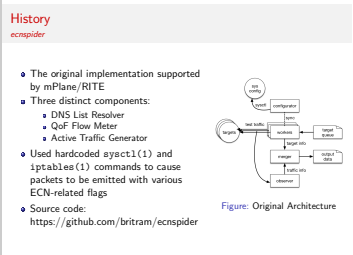
PATHspider II: The Tutorial

└History



QoF is an IPFIX Metering and Exporting process derived from the YAF flowmeter, designed for passive measurement of per-flow performance characteristics.

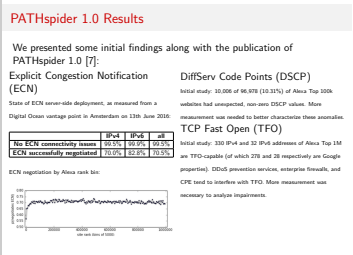
While it was fast, it was only able to export flow properties it already knew about, and could not be easily extended.

If you are interested in network measurement, you may still like to read about Internet Protocol Flow Information Export (IPFIX) [3]. It was created based on the need for a common, universal standard of export for Internet Protocol flow information from routers, probes and other devices that are used by mediation systems, accounting/billing systems and network management systems to facilitate services such as measurement, accounting and billing.

2018-06-11

PATHspider II: The Tutorial

└PATHspider 1.0 Results



Higher ranked servers tended to disable (or not support to begin with) ECN. This is likely due to the specialised nature of services that have to handle such large volumes of traffic. They may be using entirely custom codebases, or are otherwise tuned and optimised.

For a more comprehensive measurement study on the use and impairments to use of DiffServ Codepoints in the Internet, see [4].

PATHspider 2.0

- Architecture changed to add a flow combiner
- Generalised to support more than just A/B testing
  - Any permutation of any number of tests
- Replaced PATHspider's HTTP code with cURL
- Added framework for packet forging based plugins using Scapy
- Completely rewritten (in Go) target list resolver
- Observer modules usable for standalone passive observation or analysis
- Source code: https://github.com/mami-project/pathspider/tree/2.0.0/

Figure: New Architecture

The combiner thread holds a table of merged flows and waits for $|flows| = |jobs|$. Conditions are generated based on the combined flows.

Plugin Types

- Synchronised (traditional ecnspider)
  - ECN, DSCP
- Desynchronised (traditional ecnspider, no configurator)
  - TFO, H2, TLS NPN/ALPN
- Forge (new in PATHspider 2.0!)
  - Evil Bit, UDP Zero Checksum, UDP Options
- Single (new, and fast)
  - Various TCP Options

The desynchronized plugins will run more quickly than synchronized plugins while still using the real network stack. Forge plugins will run slower as there is overhead in the Scapy packet generation that doesn't exist in optimised kernel stacks.

Connection Helpers

- Instead of writing client code, use the code that already exists
- In the `pathspider.helpers` module:
  - DNS (dnslib)
  - HTTP/HTTPS (pycURL)
  - TCP (Python socket)
- For synchronised plugins, just use the helper
- For desycnhronised plugins, the helpers are customisable, e.g. cURL helpers accept arbitrary CURLOPTs

You can find the pyCURL documentation at `http://pycurl.io/docs/latest/`. This contains information on all the CURLOPTs that are currently available.

During development of PATHspider plugins, we have found that some options that exist in the C library are not included in the Python bindings, but we have been able to produce patches and upstream these relatively easily. For example: `https://github.com/pycurl/pycurl/pull/456`.

Evil Bit

The evil bit is a fictional **IPv4 packet header field** *proposed in RFC 3514 [2], a humorous April Fools' Day RFC from 2003 authored by Steve Bellovin. The RFC recommended that the last remaining unused bit, the "Reserved Bit," in the IPv4 packet header be used to indicate whether a packet had been sent with malicious intent, thus making computer security engineering an easy problem — simply ignore any messages with the evil bit set and trust the rest.*

– Wikipedia

If you enjoy the concept of the evil bit, you may like to also check out [5]: TCP Option to Denote Packet Mood. For example happy packets which are happy because they received their ACK return packet within less than 10ms. Or the Sad Packets which are sad because they faced retransmission rates greater than 20% of all packets sent in a session.