EBERHARD KARLS UNIVERSITÄT TÜBINGEN

FACULTY OF SCIENCE
Communication Networks

MalFIX: Using IPFIX for Scaling Threat Detection to High Data Rates

Gabriel Paradzik[1,2], Benjamin Steinert[1,2], Janik Steegmüller, Michael Menth[1]
[1]Chair of Communication Networks, [2]Zentrum für Datenverarbeitung
University of Tübingen

https://kn.inf.uni-tuebingen.de

Outline

Motivation

- Threat intelligence (TI) feeds provide information about indicators of compromise (IoC)
  - IoCs can be IP addresses, hostnames, signatures, etc.
  - Maintained by private companies, other network operators, or open-source projects
  - Examples: abuse.ch, AbuseIPDB

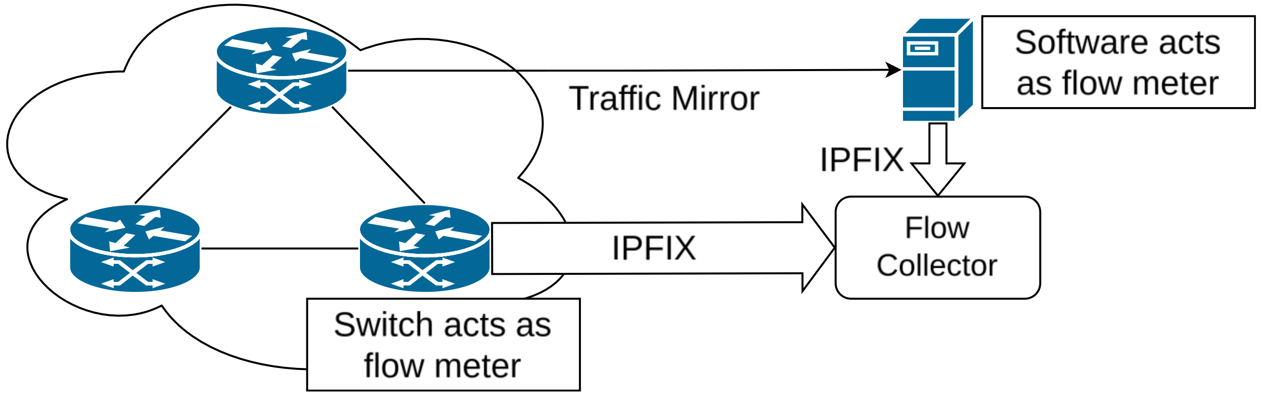
- Blocking all malicious IP addresses is unfeasible because of the large amount
  - Firewalls have limited amount of rules

- TI information can be used to identify bad actors on the network

- For networks with high volume, scanning every packet is not possible
  - Switching to flow-based scanning with IPFIX

IPFIX

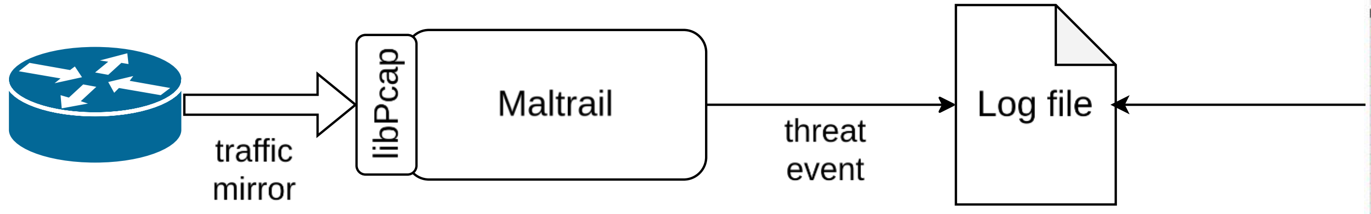
- ▶ IPFIX protocol aggregates packets into flows
  - Flow represents communication between two endpoints
- ▶ IPFIX flow record consists of multiple Information Elements (IEs)
  - IE represents certain type data point
  - Packet payload is usually discarded
- ▶ IPFIX standard allows including arbitrary data via custom IEs
  - E.g., OS/application fingerprinting, observed TCP flags

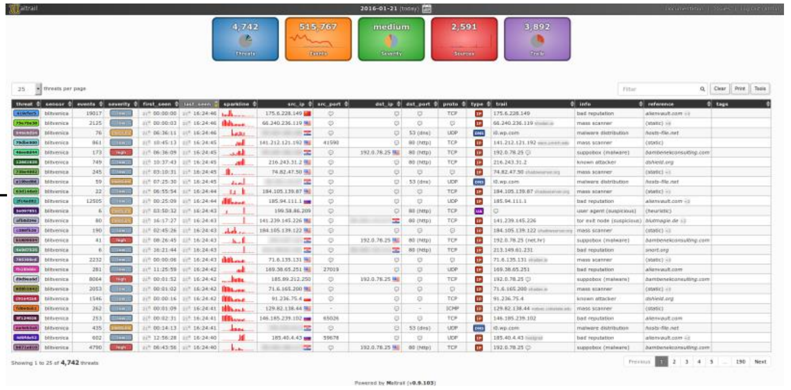


Example flow record	
flowStart	2025-03-10 14:33:25.133
flowEnd	2025-03-10 14:33:29.021
sourceIP	1.2.3.4
destIP	6.7.8.9
srcPort	44276
destPort	443
protocol	TCP
octetCount	6345
packetCount	7
tcpFlagsUnion	SYN,ACK,FIN
flowEndReason	FIN
appLabel	HTTPS

Maltrail

- Maltrail is an open-source all-in-one threat detection system written in Python
  - Actively maintained on GitHub
  - Utilizes a large number of TI feeds and static threat indicators





threat	sensor	events	severity	first_seen	last_seen	sparkline	src_ip	src_port	dst_ip	dst_port	proto	type	trail	info	reference	tags
[illegible]	blitvenica	19017	low	21st 00:00:00	21st 16:24:46		175.6.228.149				TCP	IP	175.6.228.149	bad reputation	alienvault.com	
[illegible]	blitvenica	2125	low	21st 00:00:03	21st 16:24:46		66.240.236.119					IP	66.240.236.119	mass scanner	(static)	
[illegible]	blitvenica	76	medium	21st 06:36:11	21st 16:24:46		[redacted]			53 (dns)	UDP	DNS	i0.wp.com	malware distribution	hosts-file.net	
[illegible]	blitvenica	861	low	21st 10:45:13	21st 16:24:45		141.212.121.192	41590		80 (http)	TCP	IP	141.212.121.192	mass scanner	(static)	
[illegible]	blitvenica	173	high	21st 06:36:09	21st 16:24:45		[redacted]		192.0.78.25	80 (http)	TCP	IP	192.0.78.25	suppobox (malware)	bambenekconsulting.com	
[illegible]	blitvenica	749	low	21st 10:37:43	21st 16:24:45		216.243.31.2			80 (http)	TCP	IP	216.243.31.2	known attacker	dshield.org	
[illegible]	blitvenica	245	low	21st 03:10:31	21st 16:24:45		74.82.47.50					IP	74.82.47.50	mass scanner	(static)	
[illegible]	blitvenica	59	medium	21st 07:25:30	21st 16:24:45		[redacted]			53 (dns)	UDP	DNS	i0.wp.com	malware distribution	hosts-file.net	
[illegible]	blitvenica	22	low	21st 06:55:54	21st 16:24:44		184.105.139.87				TCP	IP	184.105.139.87	mass scanner	(static)	
[illegible]	blitvenica	12505	low	21st 00:25:09	21st 16:24:44		185.94.111.1				UDP	IP	185.94.111.1	bad reputation	alienvault.com	
[illegible]	blitvenica	6	medium	21st 03:50:32	21st 16:24:43		199.58.86.209			80 (http)	TCP	UA		user agent (suspicious)	(heuristic)	
[illegible]	blitvenica	80	medium	21st 16:17:27	21st 16:24:43		141.239.145.226		[redacted]	80 (http)	TCP	IP	141.239.145.226	tor exit node (suspicious)	blutmagie.de	
[illegible]	blitvenica	190	low	21st 02:45:26	21st 16:24:43		184.105.139.122					IP	184.105.139.122	mass scanner	(static)	
[illegible]	blitvenica	41	high	21st 08:26:45	21st 16:24:43		[redacted]		192.0.78.25	80 (http)	TCP	IP	192.0.78.25 (net.hr)	suppobox (malware)	bambenekconsulting.com	
[illegible]	blitvenica	6	low	21st 16:21:44	21st 16:24:43		[redacted]		[redacted]	80 (http)	TCP	IP	213.149.61.231	bad reputation	snort.org	
[illegible]	blitvenica	2232	low	21st 00:00:06	21st 16:24:43		71.6.135.131					IP	71.6.135.131	mass scanner	(static)	
[illegible]	blitvenica	281	low	21st 11:25:59	21st 16:24:42		169.38.65.251	27019			UDP	IP	169.38.65.251	bad reputation	alienvault.com	
[illegible]	blitvenica	8064	high	21st 00:01:52	21st 16:24:42		185.89.212.250		192.0.78.25		TCP	IP	192.0.78.25	suppobox (malware)	bambenekconsulting.com	
[illegible]	blitvenica	2053	low	21st 00:01:02	21st 16:24:42		71.6.165.200					IP	71.6.165.200	mass scanner	(static)	
[illegible]	blitvenica	1546	low	21st 00:00:16	21st 16:24:42		91.236.75.4				TCP	IP	91.236.75.4	known attacker	dshield.org	
[illegible]	blitvenica	262	low	21st 00:01:09	21st 16:24:41		129.82.138.44	-		-	ICMP	IP	129.82.138.44	mass scanner	(static)	
[illegible]	blitvenica	253	low	21st 00:02:31	21st 16:24:41		146.185.239.102	65026			TCP	IP	146.185.239.102	bad reputation	alienvault.com	
[illegible]	blitvenica	435	medium	21st 00:14:13	21st 16:24:41		[redacted]			53 (dns)	UDP	DNS	i0.wp.com	malware distribution	hosts-file.net	
[illegible]	blitvenica	602	low	21st 12:56:28	21st 16:24:40		185.40.4.43	59678			UDP	IP	185.40.4.43	bad reputation	alienvault.com	
[illegible]	blitvenica	4790	high	21st 06:43:56	21st 16:24:40		[redacted]		192.0.78.25	80 (http)	TCP	IP	192.0.78.25	suppobox (malware)	bambenekconsulting.com	

Showing 1 to 25 of **4,742** threats

Previous 1 2 3 4 5 … 190 Next

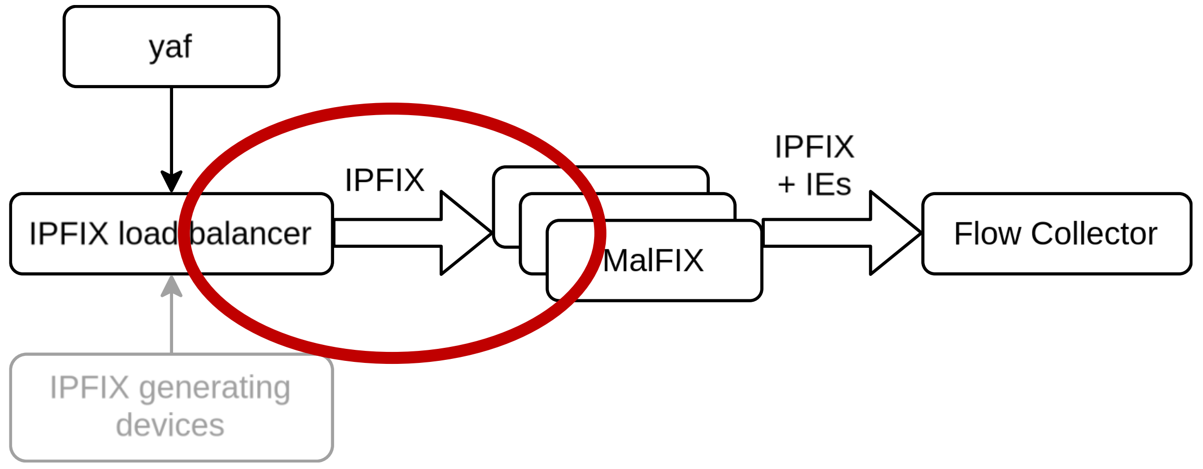
Powered by Maltrail (v**0.9.103**)

→ Perfectly suited for small networks, but not performant enough for large networks with high traffic volumes

- Can we leverage Maltrail’s up-to-date threat detection engine and use it for monitoring high traffic volumes?

MalFIX Architecture

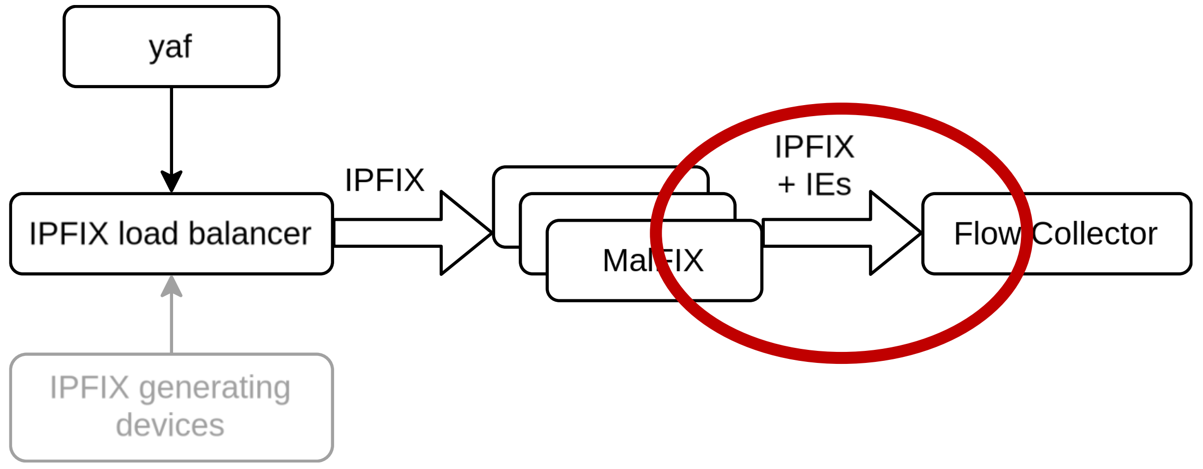
- Maltrail was modified (“MalFIX”) to allow high-performance threat monitoring
  - Changes are minimally invasive to allow easy merging with upstream
  - Input/Output capabilities were modified



- Input Adaptations
  - Instead of raw packet captures, IPFIX is accepted
  - Yaf generates IPFIX from traffic on an interface
    - High performance capturing library PF_RING™
  - Run multiple instances of MalFIX by employing IPFIX load balancer

MalFIX Architecture

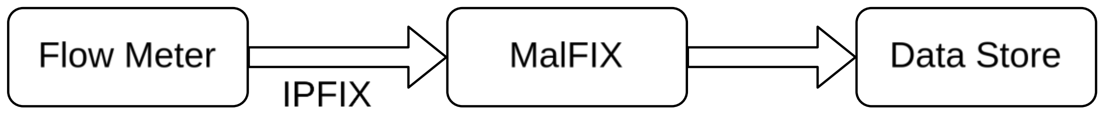
- Output Adaptations
  - Use IPFIX custom IEs
  - Detected threat information are attached via custom IEs
  - Allows for subsequent processing with IPFIX-compatible tools



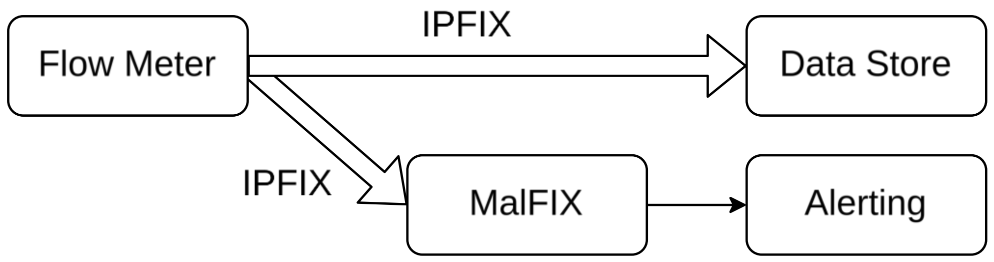
- MalFIX fully IPFIX standard compatible
  - Can be integrated into IPFIX pipelines

MalFIX: Modes of Operation

- ▶ Pipeline Mode
  - All incoming flows to MalFIX are exported
  - Custom IEs are attached to malicious flows
  - Useful for data enrichment scenarios



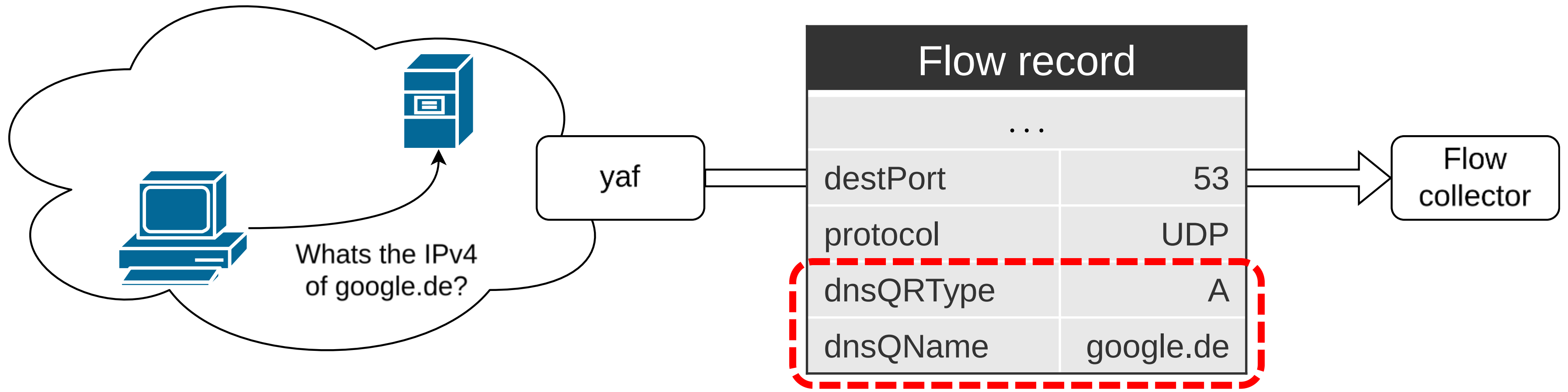
- ▶ Alert-Only Mode
  - Only malicious flows are exported
  - Non malicious flows are dropped
  - Useful for alerting



Deep Packet Inspection

- By switching from packets to flows, we lose payload information
  - Payload information is lost in typical IPFIX setup

- Yaf has Deep Packet Inspection (DPI) capabilities
  - Search for payload information (DNS, HTTP, FTP, etc.)
  - Include results in custom IEs



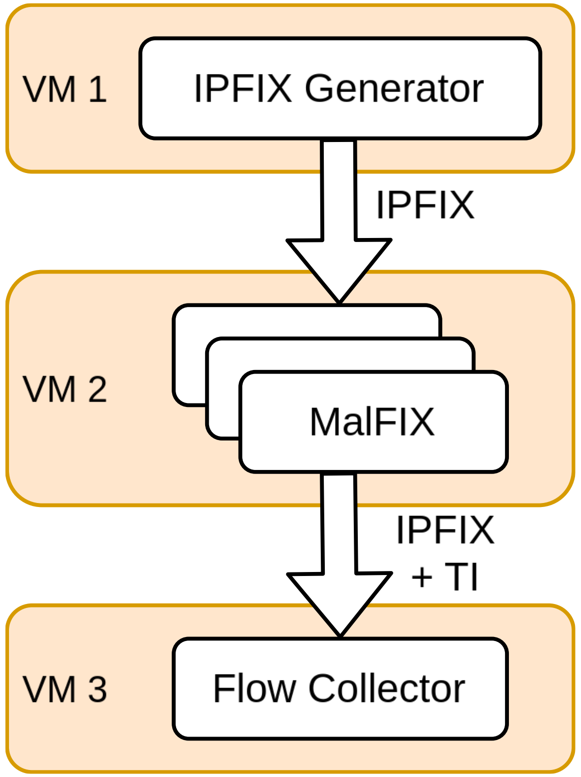
Flow record	
. . .	
destPort	53
protocol	UDP
dnsQRType	A
dnsQName	google.de

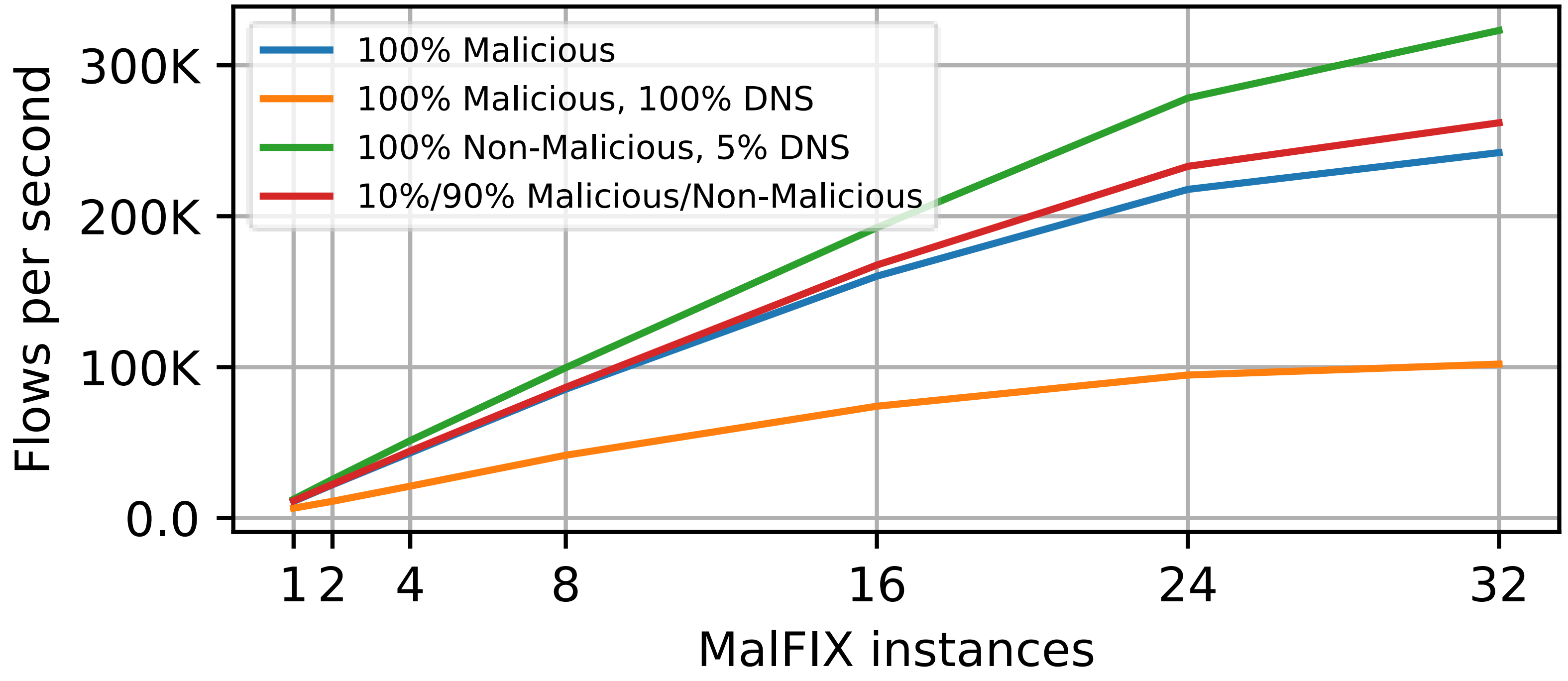
- MalFIX also reads yaf‘s DNS DPI information
  - Domain names are checked with Maltrails internal threat detection engine

EBERHARD KARLS UNIVERSITÄT TÜBINGEN

Performance Evaluation

- Maximum flow processing speed was evaluated for Alert Only Mode
- Number of running MalFIX instances was varied
- Different traffic patterns were used





Conclusion

- Open-Source tool Maltrail was modified to fit a high-performance threat detection pipeline
  - By using standard conform IPFIX, MalFIX can be integrated with other data sources/sinks
  - MalFIX is Open-source: https://github.com/uni-tue-kn/MalFIX

- Up to **300,000 flows/second** on 32 CPU cores can be scanned for threats
  - MalFIX can also be deployed across multiple machines

- MalFIX is deployed at the computation center of the University of Tübingen (ZDV)

Gabriel Paradzik[1,2], Benjamin Steinert[1,2], Janik Steegmüller, Michael Menth[1]
[1]Chair of Communication Networks, [2]Zentrum für Datenverarbeitung
University of Tübingen

gabriel.paradzik@uni-tuebingen.de

THANK YOU!

QUESTIONS?

This work was supported by the bwNET2.0 project funded by the MWK