

SHARING AND AUTOMATION FOR PRIVACY PRESERVING ATTACK NEUTRALIZATION

Graph-based Network Traffic Analysis for Incident Investigation

Milan Cermak





When some serious incident happened in the maintained network, we need to investigate its type, origin, impact, and spread to prevent further damage.

- How did the malware get on the machine?
- Did the attacker exploit any vulnerability?
- Did the machine communicate to a malware C&C or another suspicious IP address?
- Did the machine communicate with other devices in our network? How?
- Did any device from our network communicate with the same destinations as the compromised one?

Incident investigators utilize various tools to answer these questions; this presentation focuses only on network traffic analysis and specifically packet trace analysis (the same approaches are relevant for IP flow analysis and other sources of network traffic data).





- A widely-used network protocol analyzer providing insights into network activity at a microscopic level.
- De facto standard for packet trace analysis.
- + Rich and detailed support of many different protocols.
- + Ability to analyze all network traffic metadata.
- Performance issues in analyzing large packet traces.
- Limited overview of the whole packet trace.
- Missing connection to other information sources.

A	pply a display filter <0	trl-/>			Expression
No.	Time	Source	Destination	Protocol	Length Info
	4 0.025749	172.16.0.122	200.121.1.131	TCP	54 [TCP Window Update] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=
	5 0.076967	200.121.1.131	172.16.0.122		1454 [TCP Previous segment not captured] [TCP Spurious Retransmission] 10
	6 0.076978		200.121.1.131		54 [TCP Dup ACK 2#1] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=1
	7 0.102939		172.16.0.122		1454 [TCP Spurious Retransmission] 10554 → 80 [ACK] Seq=5601 Ack=1 Win=65
		172.16.0.122	200.121.1.131		54 [TCP Dup ACK 2#2] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=1
		200.121.1.131	172.16.0.122		1454 [TCP Spurious Retransmission] 10554 → 80 [ACK] Seq=7001 Ack=1 Win=65
	10 0.128319	172.16.0.122	200.121.1.131		54 [TCP Dup ACK 2#3] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=1
	11 0.154162	200.121.1.131	172.16.0.122		1454 [TCP Spurious Retransmission] 10554 → 80 [ACK] Seq=8401 Ack=1 Win=65
	12 0.154169	172.16.0.122	200.121.1.131		54 [TCP Dup ACK 2#4] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=1
	13 0.179906	200.121.1.131	172.16.0.122		1454 [TCP Spurious Retransmission] 10554 → 80 [ACK] Seq=9801 Ack=1 Win=65
	14 0.179915	172.16.0.122	200.121.1.131	тср	54 [TCP Dup ACK 2#5] 80 → 10554 [ACK] Seq=1 Ack=11201 Win=63000 Len=0
	15 0.207145	200.121.1.131	172.16.0.122	TCP	1454 10554 → 80 [ACK] Seq=11201 Ack=1 Win=65535 Len=1400 [TCP segment of
	16 0.207156	172.16.0.122	200.121.1.131	TCP	54 80 → 10554 [ACK] Seq=1 Ack=12601 Win=63000 Len=0
	17 0.232621	200.121.1.131	172.16.0.122	TCP	1454 10554 → 80 [ACK] Seq=12601 Ack=1 Win=65535 Len=1400 [TCP segment of
	18 0.232629	172.16.0.122	200.121.1.131	TCP	54 80 → 10554 [ACK] Seq=1 Ack=14001 Win=63000 Len=0
	19 0.258365 20 0.258373	200.121.1.131 172.16.0.122	172.16.0.122 200.121.1.131	TCP TCP	1454 10554 → 80 [ACK] Seq=14001 Ack=1 Win=65535 Len=1400 [TCP segment of 54 80 → 10554 [ACK] Seq=1 Ack=15401 Win=63000 Len=0
	name 15: 1454 bud	er on wine (11632 bi	ts), 1454 bytes captu	mod (11632	hite)
					:12:13 (00:0c:29:42:12:13)
			121.1.131, Dst: 172.1		,
					201, Ack: 1, Len: 1400
	Source Port: 10	554			
	Destination Por	t: 80			
	[Stream index:	0]			
	[Stream index:	n: 1400]	sequence number)		
	[Stream index: [TCP Segment Le Sequence number	n: 1400] : 11201 (relative	sequence number) lative sequence numbe	r)]	
	[Stream index: [TCP Segment Le Sequence number [Next sequence Acknowledgment	n: 1400] : 11201 (relative number: 12601 (re number: 1 (relati	lative sequence numbe ve ack number)	r)]	
	[Stream index: [TCP Segment Le Sequence number [Next sequence Acknowledgment	n: 1400] : 11201 (relative number: 12601 (re	lative sequence numbe ve ack number)	r)]	
002	[Stream index: [TCP Segment Le Sequence number [Next sequence Acknowledgment 0101 = Hea	n: 1400] : 11201 (relative number: 12601 (re number: 1 (relati	lative sequence numbe ve ack number) (5)	r)] P·\ 0·····P	ь.
003	[Stream index: [TCP Segment Le Sequence number [Next sequence Acknowledgment 0101 = Hee 0 00 7a 29 30 00 0 ff ff bc 5e 00	n: 1400] : 11201 (relative number: 12601 (re number: 1 (relati der Length: 20 bytes 50 a7 5c 30 08 e2 e 00 42 4f 78 42 56 3	lative sequence number ve ack number) (5) 2 ee bf 50 10z): 5 6a 45 52 52	P·\ 0····P ∙BO xBV5jER	R
003	[Stream index: [TCP Segment Le Sequence number [Next sequence Acknowledgment 0101 = Hee 0 00 7a 29 30 00 0 ff ff bc 5e 00	n: 1400] : 11201 (relative number: 12601 (re number: 1 (relati der Length: 20 bytes 50 a7 5c 30 08 e2 e 00 42 4f 78 42 56 3	lative sequence number ve ack number) (5) 2 ee bf 50 10z): 5 6a 45 52 52	P·\ 0····P ∙BO xBV5jER	R
003 004 005	[Stream index: [TCP Segment Le Sequence number [Next sequence Acknowledgment 0101 = Hea 0 00 7a 20 3a 00 0 ff ff b C Se 00 71 5a 69 63 39 0 61 62 46 30 77	n: 1400] : 11201 (relative number: 12601 (re number: 1 (relati der Length: 20 bytes 50 a7 5c 30 08 e2 e	lative sequence numbe ve ack number) (5) 2 ee bf 50 10 · z): 5 6a 45 52 52 6 51 34 78 35 qZic9- 7 6c 44 47 4c abFewl	P.\ 0P	R8 <5 5L

Wireshark: https://www.wireshark.org/





Network Traffic Analysis Tools – Arkime (Moloch)

- A large-scale, open-source, indexed packet capture and search tool with a web interface.
- + Indexed data storage for fast data analysis.
- + Extraction of various information from network sessions and other metadata.
- + Basic statistics of extracted data.
- + Export of selected connections as packet traces.
- No alerts correlation.
- Missing connection to other information sources.

					History Settings L							v2.7.1 ? 🤅
Search											8	🗙 Search 🧟
Last ho	ur Star	t 2021/03/01 1	.6:12:03 H H	End 202	21/03/01 17:12:03	н н е	Bounding Last Pa	acket Inter	val Auto			
									-			
0 per page	< < 1	23 > »	Showing 1 - 50 of 148	8 entries								
0				Q	Q < 10% - > S	ession Pack	ets Bytes Data byt	es Lines Bars				
0												
0												_
												_
												ll n
	1/03/01 15:00	2021/03/01 16:20:00		21/03/01 5:30:00		/03/01 40:00	2021/03/01 16:45:00	2021/03/01 16:50:00	2021/03/01 16:55:00	2021/03/01 17:00:00	2021/03/01 17:05:00	2021/03/01 17:10:00
. 🗆		Stop Time	Src IP / Country	Src Port	Dst IP / Country	Dst Po	ort ≑ Packets	Databytes / Bytes	Arkime Node	Info		
	2021/03/01	2021/03/01	93.174.93.123	46426	45.58.43.9	9569	3	0	ubuntu2004			
	17:11:08 2021/03/01	17:11:09 2021/03/01	185.156.73.57	53823	45.58.43.9	52926	3	162 0	ubuntu2004			
uch.	17:11:06	17:11:06					-	162				
	2021/03/01 17:11:05	2021/03/01 17:11:07	fe80::e86:1000:df5f:9d 00	0	ff02::1:ff00:62c6	0	3	96 258	ubuntu2004			
	2021/03/01 17:11:05	2021/03/01 17:11:07	fe80::e86:1000:df5f:9d 00	0	ff02::1:ff00:a7e3	0	3	96 258	ubuntu2004			
icmp6	2021/03/01	2021/03/01	fe80::e86:1000:df5f:9d	0	ff02::1:ff00:e2ce	0	3	96	ubuntu2004			
	17:11:03 2021/03/01	17:11:05 2021/03/01	00 2607:f170:34:10::7c0	0	ff02::1:ff00:1	0	1	258 32	ubuntu2004			
icinpo	17:11:01	17:11:01						86				
	2021/03/01 17:11:01	2021/03/01 17:11:03	fe80::e86:1000:df5f:9d 00	0	ff02::1:ff00:106c	0	3	96 258	ubuntu2004			
	2021/03/01 17:11:01	2021/03/01 17:11:03	fe80::e86:1000:df5f:9d	0	ff02::1:ff00:c440	0	3	96 258	ubuntu2004			
	2021/03/01	2021/03/01	2607:f170:34:13::670	0	ff02::1:ff00:1	0	1	32	ubuntu2004			
icinpo	17:11:01 2021/03/01	17:11:01 2021/03/01	fe80::e86:1000:df5f:9d	0	ff02::1:ff00:6d02	0	3	86 96	ubuntu2004			

Arkime: https://arkime.com/





Network Traffic Analysis Tools – Brim

- An open-source desktop application combining Wireshark and Zeek (<u>https://zeek.org/</u>) network security monitor.
- + Utilization of a Zeek to extract relevant information.
- + Indexed data storage for fast data analysis.
- + Alerts correlation (Suricata or external source).
- + Basic statistics of extracted data.
- + Export of selected connections as packet traces.
- Custom query language.
- Missing connection to other information sources.

A Brim File Edit Query View W	/indow Help	- 🗆 X
nitroba.pcap	× +	
localhost:9867 ∨ ▼ sPACES ⓑ final-all-traffic-day2.pcap ⓑ nitroba.pcap	Subspace Packets Export Columns View From	ul 22,2008 06:13:47 ▼ To ☆ 푸 : Q
V QUERIES	02 AM 02:30 03 AM 03:30 04 AM 04:30 05 AM 2008-07-22T06:13:47.046 capture_1088 814.805018 zeek 29 5,085 0.570305 All ∨ 2008-07-22T06:13:47.046 stats zeek 2,583 1,120,801 ∞ ∞ 7,893 7,889	05:30 06 AM
Activity Overview Unique DNS Queries Windows Networking Activity HTTP Requests Unique Network Connections Connection Received Data File Activity HTTP Post Requests	 2008-07-22T06:13:47.046 conn CZw9W42cis9fmyoT67 192.168.15.4 36858 74.125.11 2008-07-22T06:13:44.585 conn Cuo2Yj3UWjXImMQnJh 192.168.15.8 52675 255.255. 2008-07-22T06:13:42.009 conn Ckobuz2GcuZcQ8s95 192.168.15.8 52674 255.255. 2008-07-22T06:13:41.737 conn CexCb823TYeujZueW5 192.168.15.4 36164 84.200.22 2008-07-22T06:13:41.737 conn CDx86g2QL5kVkvby6f 192.168.15.4 36166 84.200.22 2008-07-22T06:13:41.737 conn CZ1Evq2MjtB8WKVcd 192.168.15.4 36127 84.200.22 2008-07-22T06:13:9.234 conn CpB1Eq4WbAVFVCXkw3 192.168.15.4 36822 84.200.22 2008-07-22T06:13:9.234 conn C0tjynDiwoi2nsSF6 192.168.15.4 36822 84.200.22 	9.99 80 tcp () () 255.255 2223 udp 255.255 2223 udp 27.131 80 tcp () 27.131 80 tcp () 27.131 80 tcp () 27.131 80 tcp () 27.131 80 tcp ()
▼ HISTORY Q * Q *	Linear > 2008-07-22T06:13:29.918 conn CVClZ639Q1qnucIvo2 192.168.15.4 36414 64.128.8 2008-07-22T06:13:14.997 conn CiZvcSSYg1hBFaN9 192.168.15.4 36858 74.126.19 2008-07-22T06:12:58.973 conn CYZAyF2BwTJ0jyoTVa 192.168.15.4 36858 74.125.19 2008-07-22T06:12:59.961 conn CFiyCKb689mEQMkM1 192.168.15.4 36858 74.125.19 2008-07-22T06:12:49.275 htp CJ4FcZ37wqechdMOtc 192.168.15.4 36130 17.151.16 2008-07-22T06:12:49.264 htp CJ4FcZ37wqechdMOtc 192.168.15.4 36130 17.151.16 2008-07-22T06:12:49.264 conn CJ4FcZ37wqechdMOtc 192.168.15.4 36130 17.151.16	.99 80 tcp 0 9.99 80 tcp 0 0 .99 80 tcp 0 0 .20 123 4 4 2 64 .20 123 4 3 3 64

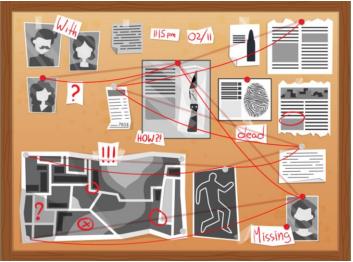
Brim: https://www.brimsecurity.com/





The human brain is used to perceiving the surrounding world and data in associations

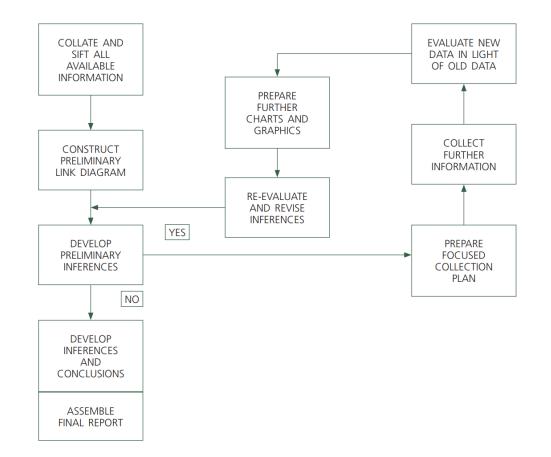
- We use associations every day, so why not use them during network traffic analysis and incident investigation?
- Traditional analysis tools provide association-based analysis only in limited form or not at all.
- Relationship diagram allows us to get a broader context to analyzed data thanks to the visual aspect.
- It is a commonly used technique in a criminal investigation.



Paper vector created by macrovector - www.freepik.com



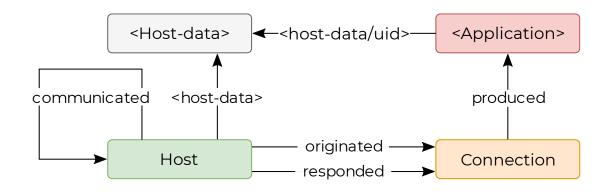




United Nations Office on Drugs and Crime (UNODC) – <u>Criminal Intelligence: Manual for Analysts</u>







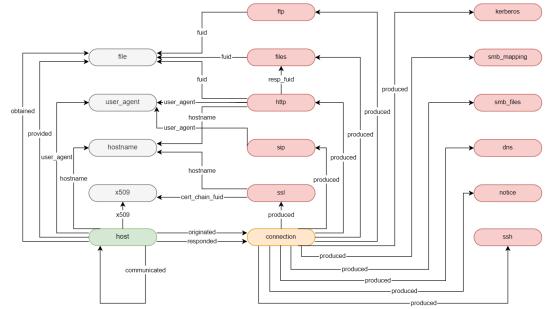
- Initial version of this representation was proposed by <u>Niese</u> and further developed by <u>Leichtnam et al</u>.
- Our model further develops these proposals and simplifies them to ease understanding by the analyst.
- Host a device with IP address observed in the network traffic capture.
- Host-data data related to the host extracted from network traffic (hostname, certificate, ...).
- Connection information about individual network connections (statistics, flags, ...).
- Application application data extracted from the connection (DNS, HTTP, TLS, ...).
- All edges should be directional to ease analysis, but reverse processing should be allowed too.





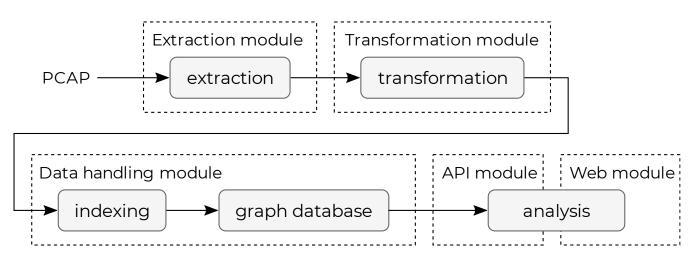
Graph-based Data Storage

- Nowadays, we can observe the rapid development of various types of databases, including graph databases that allow us to store and analyze data in the form of associations efficiently.
- Graph database examples: Neo4j (<u>https://neo4j.com/</u>), Dgraph (<u>https://dgraph.io/</u>), ...
- The graph-based approach is also used in <u>GraphQL</u>, an increasingly popular API query language.
- Utilization of a scalable database is necessary to store and analyze large-volume of network traffic data.
- For example, the dataset from the <u>CyberCzech</u> <u>exercise</u> with 330,564 connections results in 718,475 nodes and 397,632 edges.
- Current databases are better on ex-post analysis rather than continuous data storage.







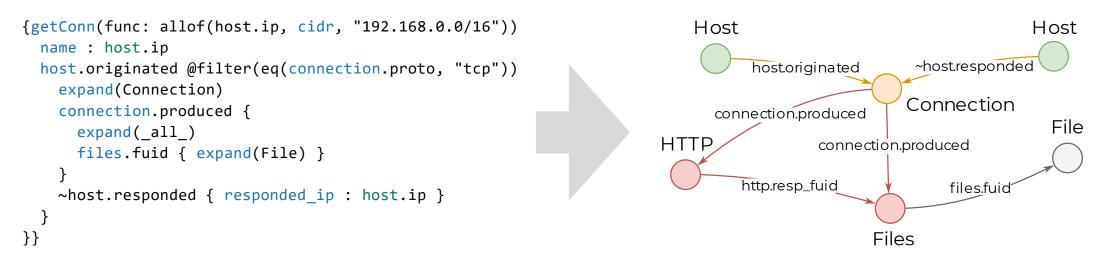


- The Granef toolkit demonstrates the presented approach to exploratory network traffic data analysis based on associations stored in a graph database.
- The toolkit's core consists of a scalable graph database Dgraph that stores transformed information from network traffic captures extracted by Zeek network security monitor.
- Modules are implemented as Docker containers.
- Custom Python scripts control all modules to ease toolkit setup and usage.
- Web interface visualizes data as an interactive relationship diagram.





 Example of a DQL query (Dgraph Query Language), based on GraphQL, with a selection of TCP connections with a file transfer from a local network:



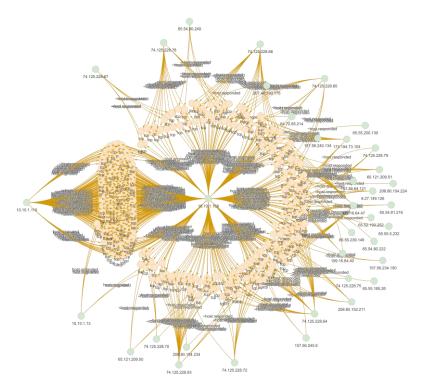
- The toolkit contains an abstract layer API with common analysis functions to ease data investigation (e.g., node neighbors' discovery, data filtering, connections overview).
- Results are provided as JSON or visualized in an interactive relationship diagram.
- Visualization uses a force-directed graph layout and allows nodes aggregation to show large relationship diagrams while preserving a simple overview of the data.





- The interactive relationship visualization allows the analyst to get details about any selected node, go into the graph's depth, and gain new observations.
- Various types of attacks and anomalies can be spotted at first glance based on visual patterns.

Granef – Graph-Based Network Forensics	\$
Custom Query Overview	
Hosts communication Hosts overview	
IP address (or CIDR): 192.168.0.0/24	
GET JSON GET GRAPH	
1/#tab-overview	







The proposed data representation allows us to easily connect other data sources and analyze them together with network traffic data within a unified environment.

- Alerts Anomalies and attacks observed by Intrusion Detection Systems can be associated either to a relevant Host node or to a specific connection.
- OSINT Data from OSINT sources can be linked to any graph node to provide a broader context.
- Host data (logs, EDR, ...) Similar graph-based representation can also be used for host-based data that can be further connected to network traffic data on a connection level.

GraphQL API allows us to obtain data from external sources directly in a format suitable for connection to a graph database. If we do not want to disturb the original data, attaching the external information only within the interactive visualization is also possible.





- Graph-based analysis follows the typical way of human thinking and perception of the characteristics of the surrounding world.
- The presented approach is not only the new method of network data storage and analysis, but it is also a shift of mindset that allows us to perceive network traffic in a new way.
- We have introduced the Granef toolkit to demonstrate exploratory network traffic analysis based on associations stored in a graph database.
- The same approach can be applied to other data types (IP flows, logs, EDRs, etc.).





Granef

We are currently finalizing the initial version of the graphical analysis environment, so the Granef toolkit is not publicly available now. But, if you are interested, just send me an email (<u>cermak@ics.muni.cz</u>) and I will be happy to provide it to you.

