Using Relational Graphs for Exploratory Analysis of Network Traffic Data

DFRWS USA 2023

Milan Cermak, Tatiana Fritzova, Vit Rusnak and Denisa Sramkova

Masaryk University, Brno, Czech Republic

Network Traffic Analysis

When an incident occurs in the network, we need to investigate its type, origin, impact, and spread to prevent further damage

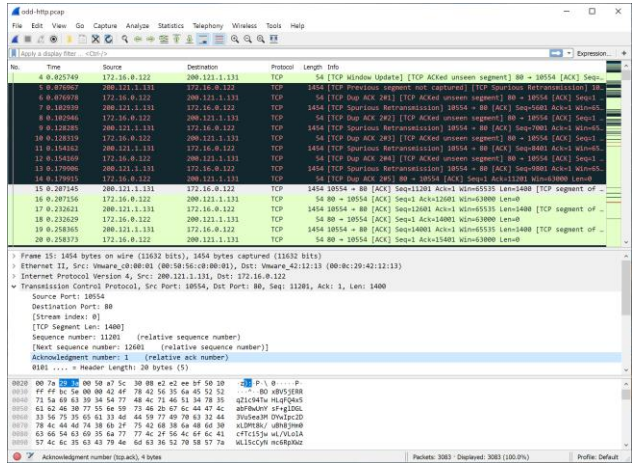
- How was the host infected?
- Did the attacker scan for open services or vulnerabilities?
- Did the host communicate to a malware C&C or another suspicious IP address?
- Did the host send a large amount of data outside the local network?
- Did the host communicate with other devices in the local network?

Incident investigators utilize various tools to answer these questions; we will focus 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)

Common Analysis Tools – Desktop

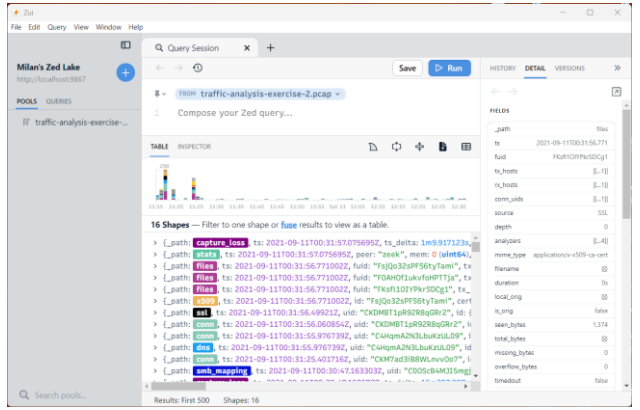
Wireshark (https://www.wireshark.org/)

- 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
- Performance issues in analyzing large packet traces



Zui (https://www.brimdata.io/)

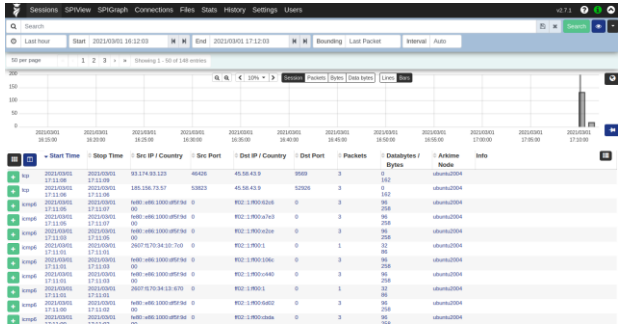
- An open-source desktop application combining Wireshark and Zeek network security monitor
- Provides indexed data storage for fast data analysis
- Utilizes custom query language



Common Analysis Tools – Web/Server

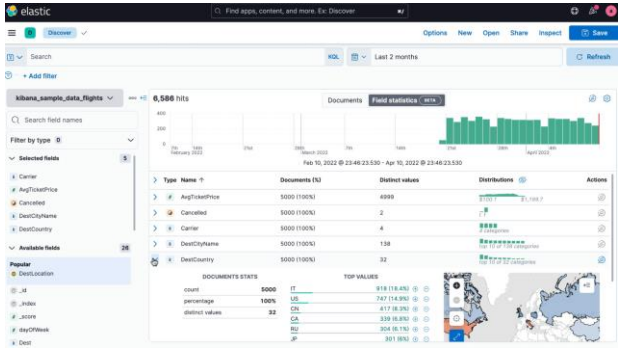
Arkime (https://arkime.com/)

- A large-scale, open-source, indexed packet capture and search tool with a web interface
- Extraction of various information from network sessions and other metadata
- Provides basic statistics of extracted data



Elastic Stack (https://www.elastic.co/elastic-stack/)

- A universal tool for the processing, storage, and analysis of various textual data (e.g., Zeek logs)
- Not directly designed for network traffic analysis
- Provides a macro-view through various visualizations of aggregated data



Data Analysis and Human Brain

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

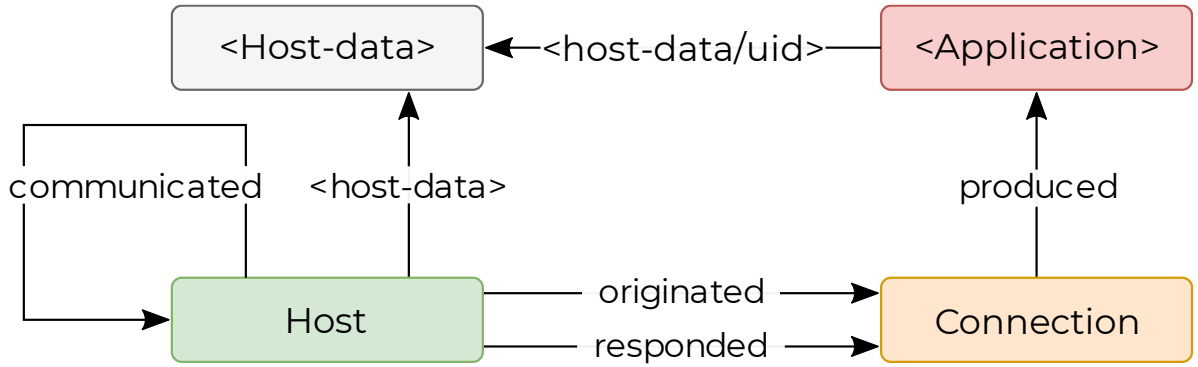
- We use associations every day, so why not use them during network traffic analysis and incident investigation?
- Traditional analytical tools provide association-based analysis only in a limited form or not at all
- Relational graph data visualization allows us to get a broader context of the analyzed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

Vector created by macrovector - www.freepik.com

Let's see how we can achieve this goal of network traffic analysis by using a graph database

- It's not perfect yet and has a lot of issues (we're working hard to resolve them)
- But the experience and the new perception of the network data are worth it!

Representation of Network Traffic Data



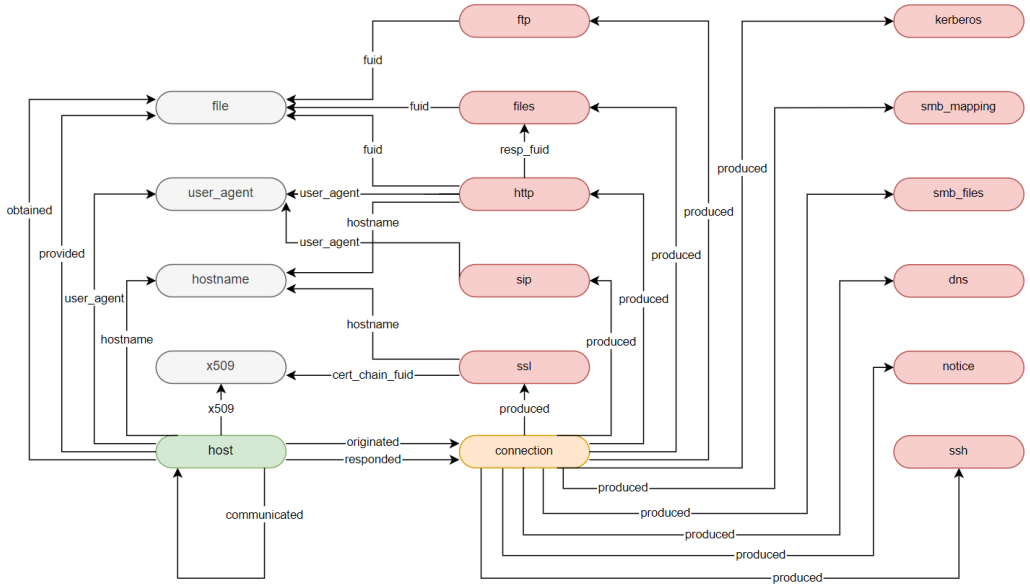
- Initial version was proposed by Niese and further developed by Leichtnam et al.
- We have further developed these proposals and simplified them to ease data understanding

- Host – a device with IP address observed in the network traffic capture
- Connection – information about individual network connections (statistics, flags, …)
- Application – application data extracted from the connection (DNS, HTTP, TLS, …)
- Host-data – data related to the host extracted from network traffic (hostname, certificate, …)
- All edges should be directional to ease analysis, but reverse processing could be possible

Graph Data Storage

- Nowadays, we can observe 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 (https://neo4j.com/), Dgraph (https://dgraph.io/), …
- The graph-based approach is also used in GraphQL, an increasingly popular API

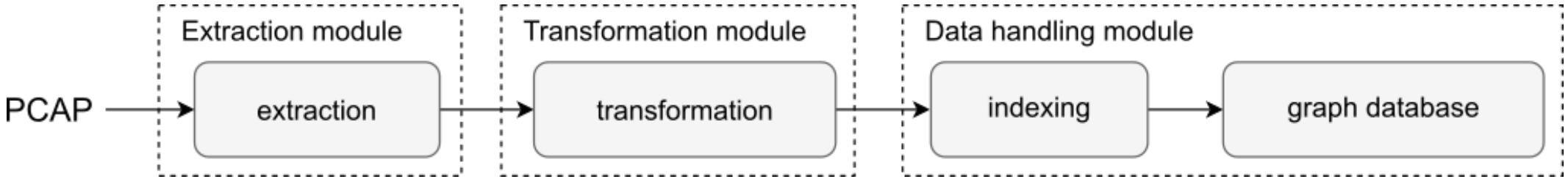
- Utilization of a scalable database is necessary to store and analyze large-volume of network traffic data
- For example, the dataset from the CyberCzech exercise with 330,564 connections results in 718,475 nodes and 397,632 edges
- Current databases are better for ex-post analysis rather than continuous data storage



Granef Toolkit

Official page: https://granef.csirt.muni.cz/

- An ETL pipeline (extract, transform, load) for processing of network traffic captures
- The core of the toolkit is a scalable graph database Dgraph (https://dgraph.io/)
- Data extraction is performed by the Zeek Network Security Monitor (https://zeek.org/)
- Data processing is performed using Docker containers
- Custom python script controls all pipeline modules to ease toolkit setup and usage
- Processes 6 GB trace file in ~7 minutes (notebook with Intel i7 @ 2.80GHz, 16 GB RAM)



Queries and Data Analysis

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

```
{getConn(func: allof(host.ip, cidr, "192.168.0.0/16")) {
  host.ip
  host.originated @filter(eq(connection.proto, "tcp")) {
    connection.ts
    connection.conn_state
    connection.produced {
      http.hostname_uri
      files.mime_type
      files.fuid { file.md5 }
    }
    ~host.responded { host.ip }
  }
}}
```


```
{ "host.ip": "192.168.1.64",
  "host.originated": [{
    "connection.ts": "2008-07-22T01:51:07.095278Z",
    "connection.conn_state": "SF",
    "~host.responded": [{ "host.ip": "74.125.19.83" }]
  },{
    "connection.ts": "2008-07-22T01:51:19.260397Z",
    "connection.conn_state": "S3",
    "connection.produced": [{
    "http.hostname_uri": "f.e.drugstore.com/i/08wk30_a2.jpg"
    "files.mime_type": "image/jpeg",
    "files.fuid": [{
      "file.md5": "ef498544b8339b30d821498f8f82b778"
    }],
    "~host.responded": [{ "host.ip": "209.3.183.2" }]
  ]}
]
```


- Once the network data is indexed, the evaluation of queries (even complex ones) is very fast
- Default web-based user interface is available at https://play.dgraph.io/

The DQL is tricky, so an interactive visual analytical interface would be better and easier to use

- We extend the ETL pipeline provided by the Granef toolkit with new modules
- The interface should provide similar analytical functionality like other common tools

Requirements

R1: Visualizing entities and their relationships

- The main attributes of the network traffic will be displayed using an oriented multimodal graph
- Selected node details can be inspected, and in-depth exploration should be possible

R2: Facilitating graph interaction

- The user will be able to customize the graph’s layout and other interface elements

R3: On-demand data enrichment

- The analyst can enrich network traffic data with additional information from external sources

R4: Visual and parametric filtering

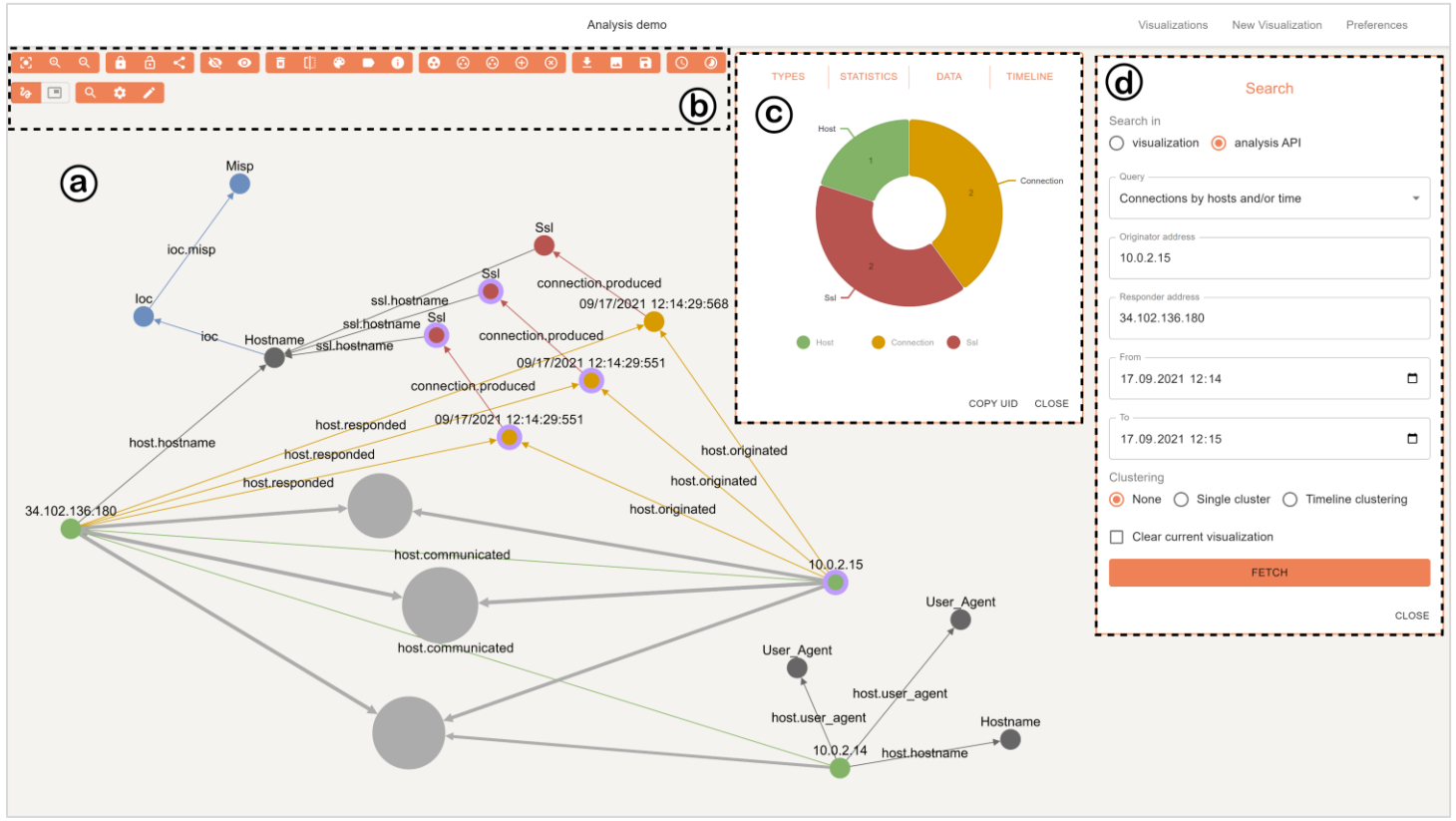
- The data selection will be possible by entering into a form and by direct graph interaction

R5: Scalability

- The system must be able to display graphs with thousands of graph nodes

Visual Analytics Interface

Combination of API using the Dgraph database and web-based interface provides functionality for exploratory analysis of stored network data



ⓐ View with rendering oriented relational graph

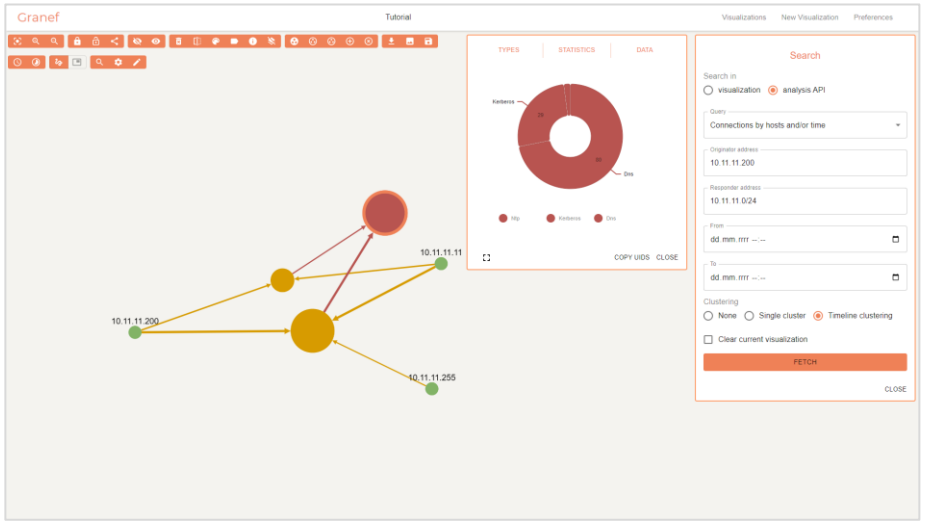
ⓑ Tools menu providing different options to interact with the graph

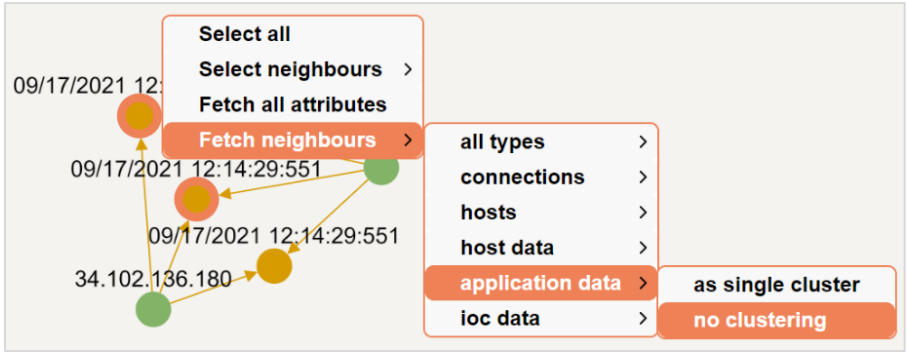
ⓒ Detail child window providing various details related to the selected node(s) or edge(s)

ⓓ Search dialog for structured filtering and data querying

Visual Analytics Interface – Graph View

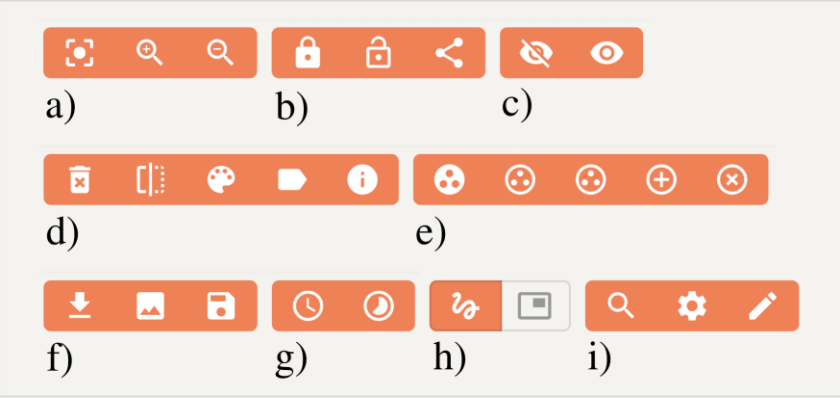
- An interactive representation of an oriented graph that matches the visual encoding (same as the schema) of graph nodes and relations between them (**R1**)
- When loaded, the graph layout algorithm provides an initial positioning
- Users can interact with the chart through the context menu that is invoked by a right-click
- To overcome performance degradation with the growing number of graph elements, the nodes can be gathered into node clusters (**R5**) – based on time, centrality, or manually

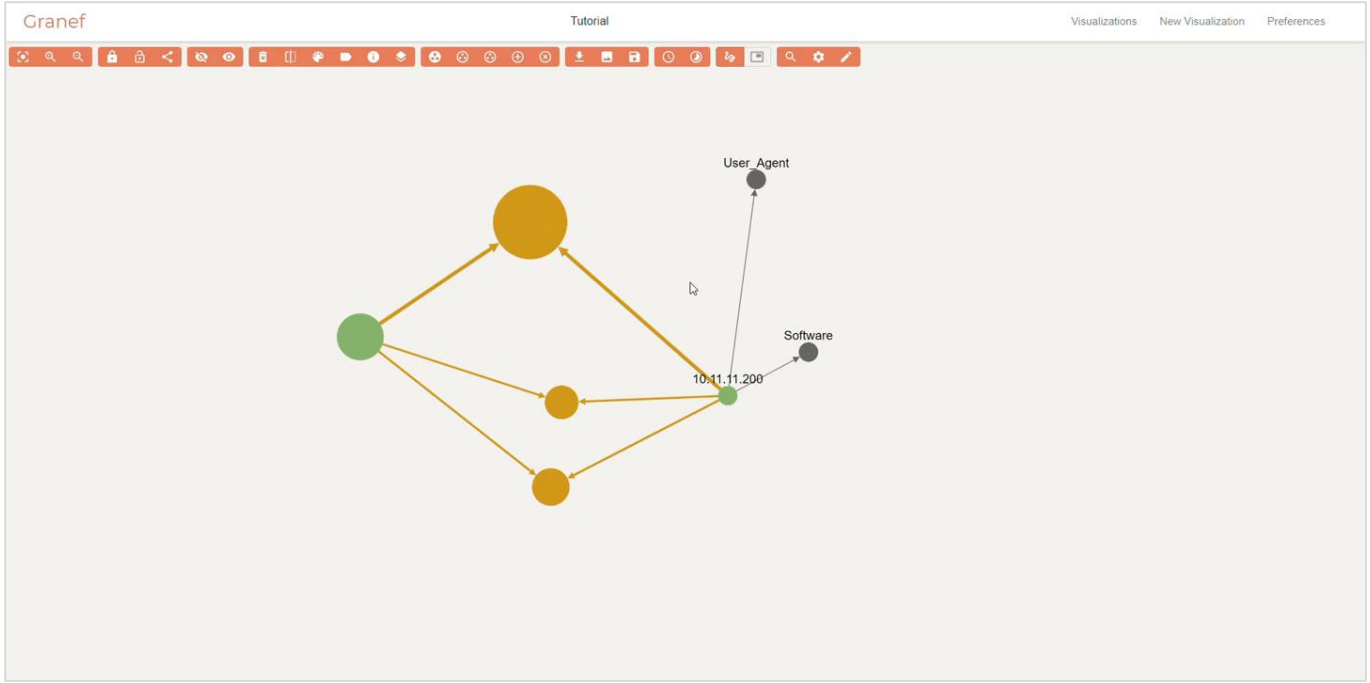




Visual Analytics Interface – Tools Menu

- Provides direct access to the most frequent interaction tasks via action buttons on top (**R2**)
- **Actions are grouped into nine categories:** **a)** view manipulation, **b)** node locks, **c)** node hiding, **d)** graph actions, **e)** clustering actions, **f)** export and save, **g)** timeline controls, **h)** selection mode, **i)** other

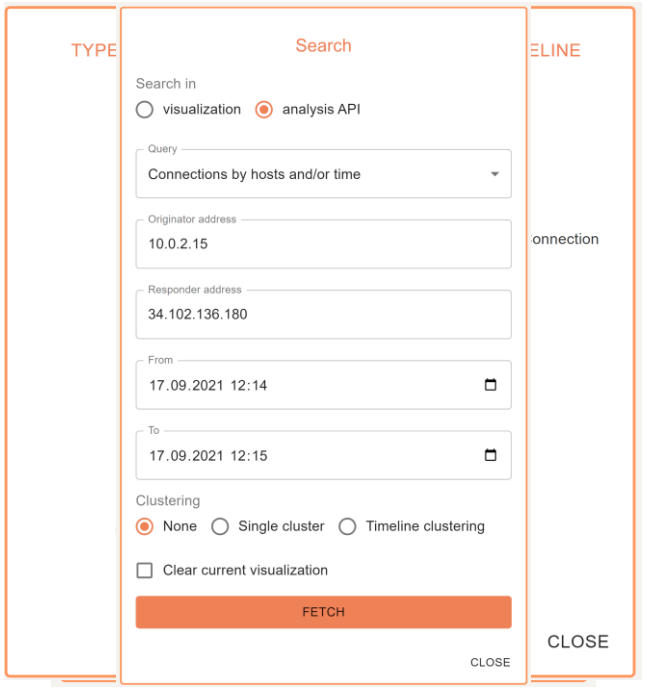


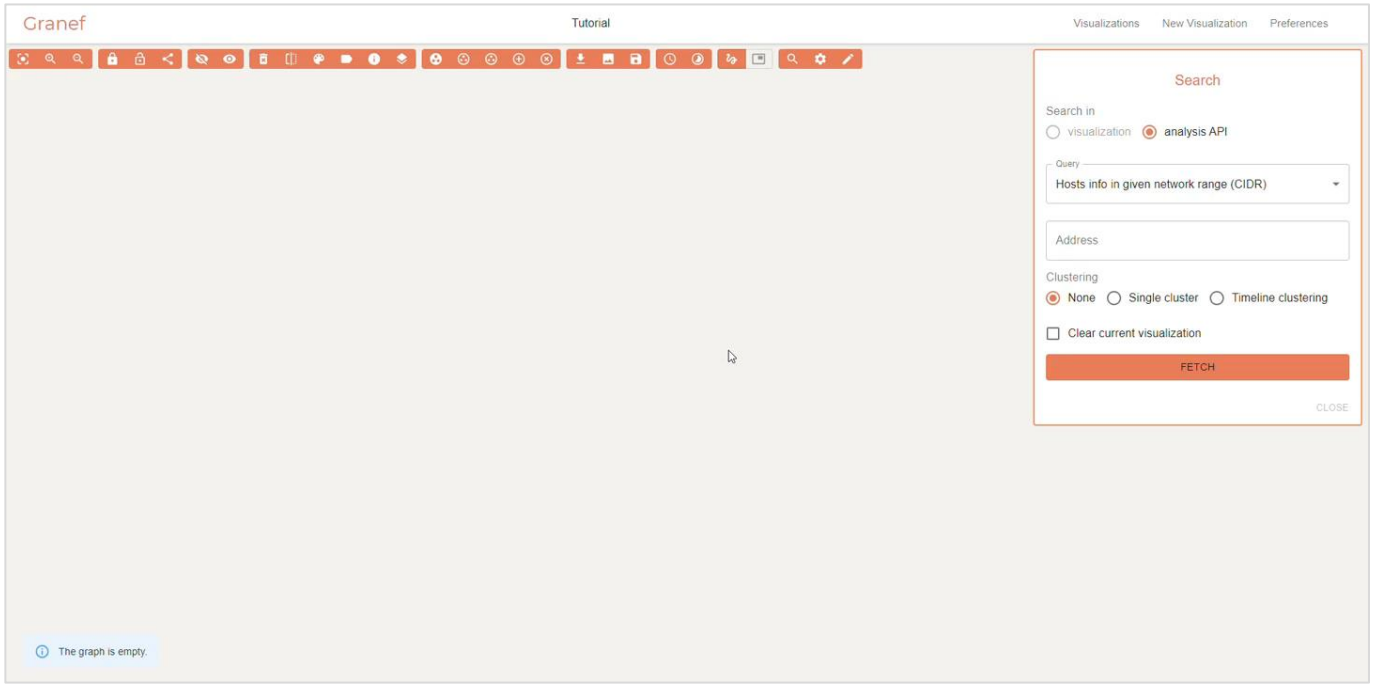


Visual Analytics Interface – Child Windows

The interface provides three types of child windows:

- Detail – offers additional information related to the selected nodes (**R3**)
- Search – allows the user to filter the data using parametric querying (**R4**)
- Timeline – allows the user to filter the data based on connection time





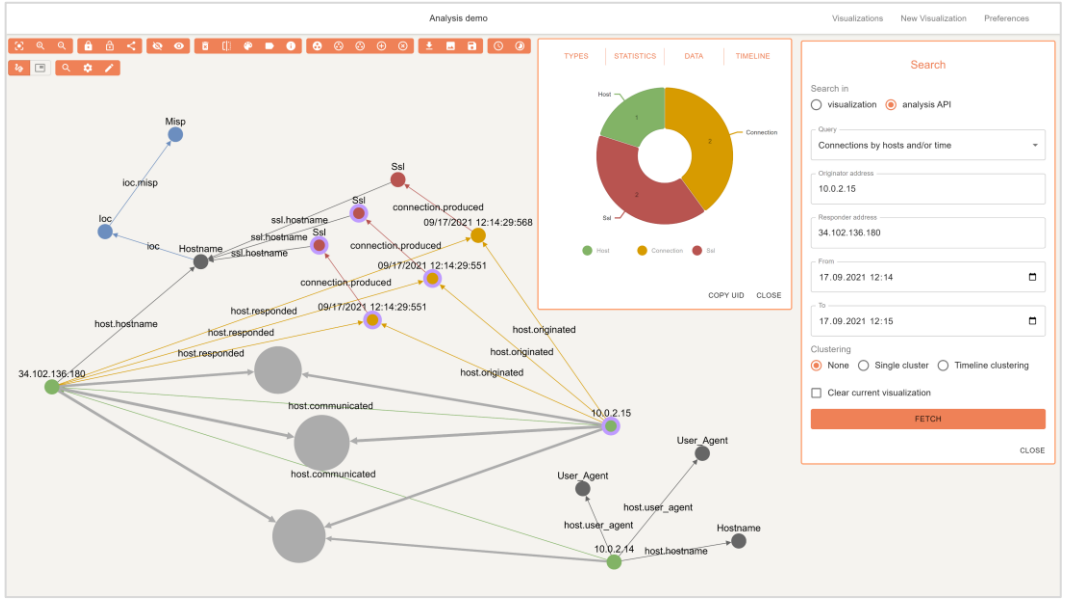
It is a nice analytical interface, but does it work in real cases?

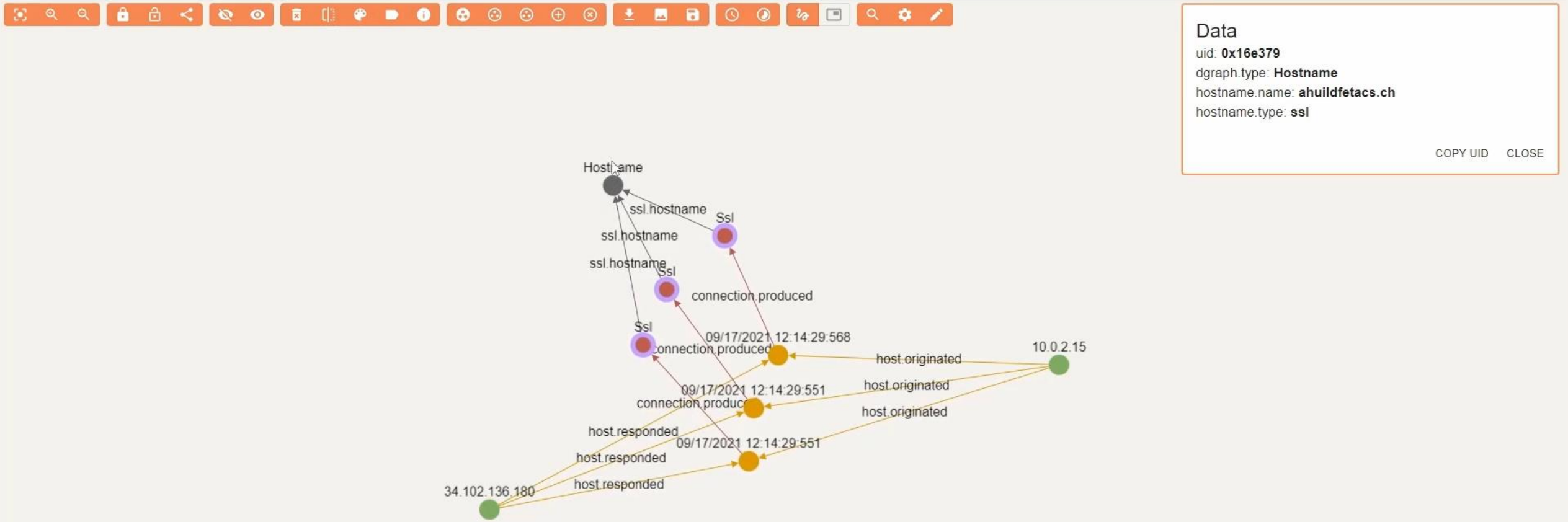
- We have prepared sample solutions for two realistic analytical cases
- We conducted a user study with domain experts from a CSIRT team

Analysis of Malicious Domain Connection

- Realistic scenario based on the SAPPAN dataset containing network traffic from a local network with multiple infected hosts communicating with the command-and-control center
- Extended with the threat intelligence data describing IoCs related to the used malware
- The analysis starts with an IDS alert identifying communication with a malicious domain (the analyst knows related IP addresses and connection time)


```
{
  "@timestamp": "2021-09-17T12:14:31.812000Z",
  "ipfix:sourceIPv4Address": "10.0.2.15",
  "ipfix:destinationIPv4Address": "34.102.136.180",
  "domain_hostname": "xczjhkgdsadsa.ch",
  "alerts": {
    "10438f5c-aea7-4cca-8ab9-641316f56a15":
    "http.domain.dga.score > 0.5"
  },
  "enrichment": {
    "domain_hostname": {
      "freq_score": 3.4317,
      "dga": {
        "score": 0.997,
        "family": "locky"
      }
    }
  }
}
```



Data

uid: **0x16e379**
dgraph.type: **Hostname**
hostname.name: **ahuildfetacs.ch**
hostname.type: **ssl**

COPY UID CLOSE

User Study

Study scenario and characteristics

- Based on the CSE-CIC-IDS2018 dataset (part Thurs-22-02-2018)
- Aims to verify the identified incident and whether the attacker performed any other attacks
- Five cybersecurity data analysis experts with experience between 5 and 12 years
- Provided feedback and filled out a System Usability Scale (SUS) questionnaire

Results

- After the initial confusion, each expert verified alert correctness, analyzed related network traffic, and identified the two additional attacks
- SUS score: 78 – Acceptable or Good (B+) in adjective interpretation or the numeric values
- The biggest difficulty for participants was unfamiliarity with the data model
- A positive surprise for us was that the participants were able to use clusters intuitively, which allowed them to get an overview of the contained data quickly

Summary and Conclusion

Conclusion

- 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 extended an open-source Granef toolkit (https://granef.csirt.muni.cz) by new modules providing visual interface supporting exploratory analysis of network traffic data
- The evaluation showed that once the analysts got used to the new data model, they could quickly investigate the incident and reveal all necessary information

You can try using the Granef toolkit and the new interactive visual analytical interface by following the tutorial at https://granef.csirt.muni.cz/tutorial/

Check granef.csirt.muni.cz to get more information about Granef and our research!

Feel free to contact me also at cermak@ics.muni.cz