Development and Analysis of a Reconnaissance-Technique Knowledge Graph

Elsa Deitz[1], Eve Cohen[1], Jordana Wilkes[1] and Thomas Heverin[2]
[1]The Baldwin School, Bryn Mawr, United States of America
[2]Girls Learn Cyber, Inc., Philadelphia, United States of America

hacker@girlslearncyber.com
thomas@girlslearncyber.com

Abstract:Penetration testing involves the use of many tools and techniques. The first stage of penetration testing involves conducting reconnaissance on a target organization. In the reconnaissance phase, adversaries use tools to find network data, people data, company/organization data, and attack data to generate a risk assessment about a target to determine where initial weaknesses may be. Although a small number of tools can be used to conduct many of reconnaissance tasks, including Shodan, Nmap, Recon-ng, Maltego, Metasploit, Google and more, each tool holds an abundance of specific techniques that can be used. Furthermore, each technique uses unique syntax. For example, Nmap holds over 600 scripts that make up its Nmap Scripting Engine. Depending on the type of device targeted, Nmap scripts can scan for ports, operating systems, IP addresses, hostnames and more. As another example, Maltego operates over 150 transforms or modules that collect data on organizations, files and people. Understanding which reconnaissance tool, techniques within those tools, and the syntax for each technique represents a highly complex task. MITRE ATT&CK, a widely accepted framework, models tactics and techniques within the tactics to help users make sense of adversarial behaviours. The tactic of reconnaissance is modelled in ATT&CK as well as its techniques. However, the explicit links between reconnaissance techniques are not modelled. Our research focused on the development of an ontology called Recontology to model the domain of reconnaissance. Recontology was then used to form Reconnaissance-Technique Graph (RT-Graph) to model 102 reconnaissance techniques and the directional links between the techniques. We used exploratory data analysis (EDA) methods including a graph spatial-layout algorithm and several graph-statistical algorithms to examine RT-Graph. We also used EDA to find critical techniques within the graph. Patterns across the results are discussed as well as implications for real-world uses of RT-Graph.

Keywords: Knowledge Graph, Ontology, Reconnaissance

1. Introduction

MITRE ATT&CK serves as a model of adversary tactics and techniques that come from real-world observations. It represents a widely accepted knowledge base of adversary behaviours. ATT&CK has been widely accepted in both research and industry domains and has been applied to adversary emulation, red teaming, security tool development, and security operations center (SOC) assessments (Georgiadou, Mouzakitis, and Askounis, 2021). ATT&CK consists of 14 tactics that include Reconnaissance, Initial Access, Execution, Command and Control, Exfiltration, and others. Each tactic contains various techniques that adversaries use to achieve the tactic's goal.

We focused our research on the Reconnaissance Tactic of ATT&CK. MITRE (2022) defines reconnaissance as "…techniques that involve adversaries actively or passively gathering information that can be used to support targeting." Reconnaissance techniques (RTs) are used to collect data such as device names, IP addresses, host names, employee names, email addresses, and more. MITRE (2022) stated that information collected from one RT can be used "to drive and lead further reconnaissance techniques." Although ATT&CK conceptualizes that one RT can inform or lead to other RTs, ATT&CK does not explicitly model these technique-to-technique connections. Our research aimed to show how a knowledge graph could model direct connections among RTs and to show how graph statistical algorithms could be used to explore the data within the knowledge graph. Results and insights are provided about the graph statistical analyses run over the knowledge graph.

2. Background

The reconnaissance domain represents a difficult domain to understand due to various challenges. These challenges include navigating an abundance of available tools, figuring out the varied uses of each tool, sifting through various types of data produced by the tools, and understanding the relationships among all the data (Steinberg, 2017). Ontologies serve as one way to help reduce the complexity of these challenges. Ontologies model domain concepts and relationships among the concepts. Gruber (1993, p. 1) defined an ontology as "…a representational vocabulary for a shared domain of discourse — definitions of classes, relations, functions, and other objects." People can gain an understanding of a domain from ontology. Ontologies can be used to create a body of knowledge about a domain. For example, Aviad, Wecel, Abramowicz (2015) used an ontology to model the cybersecurity body of knowledge.

An ontology's modelled concepts (also called classes) provide "data slots" for specific objects from a domain. For example, in the ATT&CK Reconnaissance domain the objects are specific RTs. An ontology's defined relationships among domain concepts also provide "slots" for named linkages between objects. As an example, the RT of finding someone's name on a device can be linked to the RT of searching for that name on LinkedIn.

When an ontology is populated with specific objects and explicit links are made between the objects, the ontology becomes a knowledge graph. Kurniawan et al. (p. 374, 2022) defined a knowledge graph as "…a directed, edge-labelled graph $G = (V, E)$ where V is a set of vertices (nodes) and E is a set of edges (properties). A single graph is usually represented as a collection of triples $T = < s\ p\ o >$ where s is a subject, p is a predicate, and o is an object." Each triple represents an object being linked to another object via a predicate. The predicate is also known as an object property.

Ontologies have been used in the cybersecurity domain for knowledge graph construction. Examples include the Situation and Threat Understanding by Correlating Contextual Observations (STUCCO) ontology (Iannacone et al., 2015) and the Unified Cyber Ontology (UCO) (Syed et al., 2016). Many open-source libraries also contain ontologies for the knowledge graph construction (Liu et al., 2022). Many types of graphs have been developed in the cybersecurity domain. As an example, Noel et al. (2015) developed a graph that predicts possible attack paths based on attack patterns and defense technologies. Ye et al. (2019) created a graph focused on attack paths to improve vulnerability assessments. There also have been many graphs developed that focus on security assessments (Zhang and Liu, 2020) and network security analytics (Noel, 2018).

Kurniawan et al. (2022) and Liu et al. (2022) identified a research need after examining reviews of previous cybersecurity knowledge-graph studies (Ding, Liu, and Zhu, 2020; Dong et al., 2020; and Yan and Liu, 2020). The identified research need consists of showing how knowledge graphs can be used to practical problems in cybersecurity at a broader scale than on a specific type of network with specific types of devices. Our research aimed to address this need by demonstrating a way to add to the Reconnaissance Tactic of ATT&CK. We first developed an ontology (called Recontology) and then filled the ontology with 102 RTs to form RT-Graph. Our research analyzed RT-Graph with statistical methods to draw insights about RTs. Using quantitative methods over graphs is considered to be critical for graph analyses (Noel et al., 2016).

3. Recontology and RT-Graph Development

3.1 Ontology Data Sources

Various sources can be used to create ontologies including industry-accepted frameworks, textbooks, standards, reports, domain expertise and others. For Recontology development, we used both ATT&CK, an industry-accepted framework in cybersecurity, and our domain expertise. Sugumaran and Storey (2002) considered domain expertise to be one of the most critical sources for ontology development. We hold over 10 years of experience in building cybersecurity ontology models and knowledge graphs in U.S. defense domain, 6 years of active-duty military experience, and multiple cybersecurity certifications including the Certified Information Systems Security Professional (CISSP) certification. We also have earned recognition for our reconnaissance work. Evidence includes being named on a university bug bounty program, having multiple Google Hacks accepted on the Google Hacking Database (GHDB), having submitted security reports to various university security teams, and having submitted a Common Vulnerabilities and Exposures (CVE) report to the National Vulnerability Database (NVD) which is pending review. In our reconnaissance work we have used various tools including Shodan, Nmap, Google, GHDB, Maltego, Metasploit and several more. We also use each tool in a variety of different ways.

3.2 Ontology Structure

Ontologies contain three main entities: classes, object properties, and data properties. Classes represent categories of objects. Object properties link together classes and data properties describe class characteristics. We used Protégé, an open-source ontology development tool, to create Recontology. Figure 1 shows Recontology classes. We created a super-class called Reconnaissance Tactic which contains all the RT classes. This Reconnaissance Tactic super-class maps directly to the ATT&CK Reconnaissance Tactic.

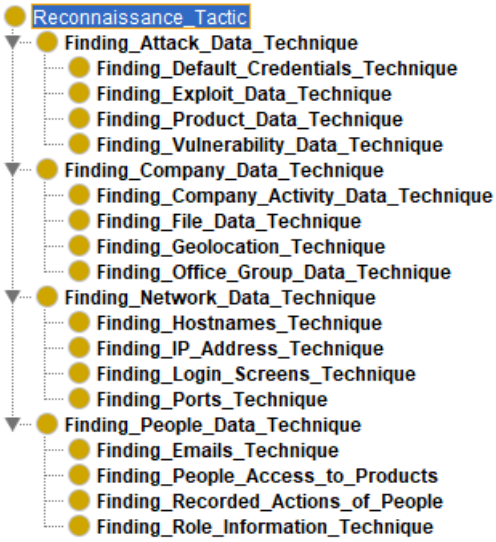

Figure 1: Classes in Recontology as Represented in Protégé

There are four upper classes that fall under the super-class Reconnaissance Tactic. The four upper classes are: Finding Attack Data Technique, Finding Company Data Technique, Finding Network Data Technique, and Finding People Data Technique. These four upper classes came from our years of experience in reconnaissance. Each upper-class contains sub-classes. There are a total of 16 sub-classes. Examples include Finding Default Credentials, Finding Hostnames, and Finding Emails techniques.

Recontology currently contains one object property called *providesDataFor*. We used this object property to show how results from one technique can provide data for another technique which is what we experience in the reconnaissance domain. This object property also directly relates to MITRE's (2022) conceptualization that RTs can "reveal opportunities for other forms of reconnaissance." In Recontology we explicitly model the connection among RTs.

Data properties provide an opportunity to name metadata about objects within classes. In Recontology, we use the following data properties: MITRE ATT&CK ID, Software, and Use. The MITRE ATT&CK ID data property is used to map Recontology RTs to MITRE ATT&CK technique ID numbers. The Software data property is used to define what tool is used for a specific Recontology RT. The MITRE (2022) definition of software is: "a generic term for custom or commercial code, operating system utilities, open-source software, or other tools…" The data property called Use provides a slot for stating the exact syntax to use for each RT.

Once objects were placed into the Recontology and were explicitly linked together via the object property *providesDataFor*, Recontology became RT-Graph. We used Protégé to enter the RTs and to explicitly link them together with direction (a RT points to another RT). When direction is modelled in a graph, the graph becomes a "directed graph." Across the 14 Recontology subclasses shown in Figure 1, there are 102 specific RTs within RT-Graph and the object property *providesDataFor* is specified over 200 times. We also filled out the three data properties (MITRE ATT&CK ID, Software, and Use) for each RT.

Figure 2 shows example RTs contained within the Finding Emails Technique class of RT-Graph.

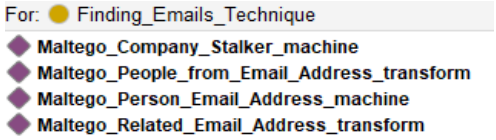

Figure 2: Objects Found in the Finding Emails Technique Class of RT-Graph

Figure 3 shows an example of the object property *providesDataFor* specified in RT-Graph. The example focuses on a specific Maltego RT.

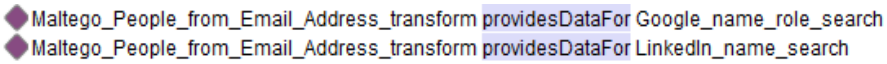

Figure 3: Example of the object property providesDataFor used in RT-Graph

After the ontology was developed in Protégé and the data for RTs (as well as object and data properties for each RT) were entered to form RT-Graph, we used the Protégé OWL2GraphML plug-in to transform the Web Ontology Language (OWL) file to a GraphML file. The OWL file format is a standard format used to save an ontology in Protégé and the GraphML file is an XML-based file that can be used by various graph-analysis programs such as Gephi.

4. Exploratory Data Analysis

We used exploratory data analysis (EDA) techniques to analyze the 102 RTs and the relationships among the RTs found in RT-Graph. With EDA, researchers attempt to find patterns and draw insights with data visual analytics, such as graph spatial-layouts, and with graph statistical algorithms (Langer and Meisen, 2019). Gephi, a graph visualization tool, provides various spatial layout algorithms and graph statistical algorithms for EDA.

4.1 Descriptive Metrics about the Graph

Initially, we focused on RT-Graph class and data property metrics. More specifically, we focused on the counts of RTs found in each of the 14 subclasses and the counts of software (such as Shodan, Nmap, Recon-ng and more) being used. We then examined trends across these metrics.

4.2 Spatial Algorithms for Visual Analysis for the Graph

For a directed graph like RT-Graph, there are various layout spatial-layout algorithms one can use within Gephi such as ForceAtlas, Fruchterman-Reingold, and Yifan Hu algorithms. Spatial layouts allow one to get a sense of groups of nodes (also known as subgraphs) and their characteristics, nodes that link subgraphs together, flow of information across graphs (as represented by arrows), and more. We chose the Force Atlas algorithm which is primarily used for directed graphs that have less than 10,000 nodes. Additionally, researchers can select filters to only show selected nodes based off statistical measures.

4.3 Statistical Data Analysis of Graph Features with Algorithms

Gephi provides several graph-statistical algorithms for data analyses. We explored the use of various algorithms in Gephi and selected the most appropriate ones for RT-Graph. The selected algorithms are named in Table 1 along with the measures they produce and meanings of the measures.

Table 1: Graph Statistical Algorithms

Algorithm Name	Algorithmic Measure	Explanation of Algorithmic Measure Results
Average Weighted Degree Algorithm	Weighted In-Degree	Weighs the number of incoming connections to a node; the higher the number, the more incoming connections
Average Weighted Degree Algorithm	Weighted Out-Degree	Weighs the number of outgoing connections; the higher the number the more outgoing connections
Average Weighted Degree Algorithm	Weighted Degree	Weighted total of incoming and outgoing connections; the higher the number the more "traffic" going through a node
Network Diameter Algorithm	Betweenness Centrality	The higher the number, the more critical the node is in the graph
Eigenvector Centrality Algorithm	Eigenvector Centrality	The higher the number, the more important a node is to the graph
Hyperlink-Induced Topic Search (HITS) Algorithm	Authority	The higher the number, the more valuable information the node holds
Hyperlink-Induced Topic Search (HITS) Algorithm	Hub	The higher the number, the more the node points to valuable, authoritative nodes

We analysed the results of each algorithm to look for trends within each algorithmic measure. We then compared the results across the algorithms to also look for trends. We also examined results across the descriptive graph metrics, spatial algorithms results, and statistical algorithms to draw insights and to examine patterns across the RT data.

5. Exploratory Data Analysis Results and Insights

5.1 RT-Graph Descriptive Metrics

RT-Graph consists of 16 subclasses that contain specific RTs. Examination of the RTs across the classes showed the following types of classes have the most RTs: finding vulnerability data (17), finding hostname data (14), finding IP address data (14), and finding port data (11). We also found there are a considerably smaller number of RTs for finding default credentials (2), log in screens (2), office-group data (1), and recorded actions of people techniques (1).

We examined the values of the data property "Software" for all 102 RTs, tracked the counts, and sorted the counts in Table 2.

Table 2: Counts of Tools Corresponding to RTs in RT-Graph

Shodan - 25	Device Access - 18	Nmap - 16	Recon-ng - 15
Google - 10	GHDB - 6	Maltego - 6	Metasploit - 2
Exploit DB - 1	LinkedIn - 1	NVD - 1	SpeedGuide - 1

We found that Shodan appeared the most (25) for all RTs. Upon examination of the Shodan RTs, we found that many of the techniques focus on finding a wide variety of data including screenshots, favicons, products, IP addresses, vulnerabilities, hostnames, ports, image, locations, and text in HTTP titles. We also found that a lot of the Shodan RTs rely on using hostnames, IP address or organizational names to find the aforementioned data. It was not surprising to use to see Shodan having the highest count among the tools. We use Shodan quite often in our reconnaissance work.

We also analyzed the access methods to the tools. Tools like Shodan, Device Access (meaning logging onto a device with credentials), Google, GHDB, ExploitDB, LinkedIn, the NVD, and SpeedGuide can all be used from a browser. However, some do require an account to be set up (Shodan, Device Access, and LinkedIn). We also found that 63 out of 102 tools (61.8%) can be accessed directly and freely from a browser while 38.2% require an install or download. Tools requiring an install include Nmap, Recon-ng, Maltego, and Metasploit. The above shows that our experience relies heavily on browser-based tools that are freely accessible. Also, all tools in RT-Graph are open source and free to use. Although these above results provide insights about the counts of data in RT-Graph, the descriptive results do not provide a clear picture about the relationships of and influence of RTs in RT-Graph.

5.2 Spatial Algorithm Results

Figure 4 shows the ForceAtlas spatial layout for RT-Graph. The parameter Weighted Out-Degree (described in Table 1) was selected to show the weighted directional-flow from RT to RT.

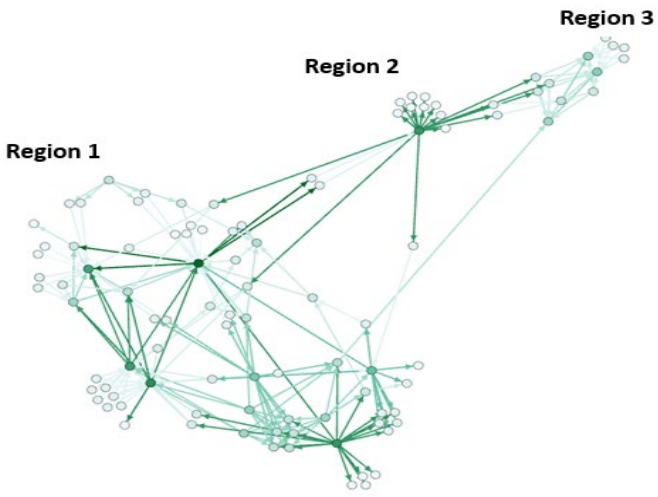

Figure 4: ForceAtlas Algorithm Spatial Layout of RT-Graph

The overall ForceAtlas spatial layout shows three large sub-graphs (or group of nodes). We labelled these as Regions 1, 2 and 3. In Region 1, many of these nodes are RTs that focus on finding things vulnerability, hostname, IP address and port data. This is the largest subgraph of RT-Graph. This made sense to us as the counts of RTs for finding these types of data are highest in RT-Graph as stated in Section 5.1

Region 2 represents both accessing a device and then searching the device for more data to use in other RTs. Finally, Region 3 represents searching for people data such as email addresses, access lists, and work roles. One can see also that some nodes sit in between Regions 1 and 2 that point /from various regions. While the spatial layout provided an overview of the whole RT-Graph, graph statistical-algorithms were needed to understand more details about RT-Graph.

5.3 Graph Data Analysis Results and Interpretation

Table 3 shows the results for Weighted In-Degree, Weighted Out-Degree, and Weighted Degree measures calculated by the Average Weighted Degree algorithm.

Table 3: Weighted In-Degree, Out-Degree and Degree Results for RT-Graph

Algorithmic Measure (Maximum Value for RT-Graph)	Top Results From RT-Graph (Measure Value)
Weighted In-Degree (Maximum = 20.0)	• Google search for product name (20.0) • ExploitDB search for exploits (17.0) • Nmap Intense scan for all TCP ports (15.0)
Weighted Out-Degree (Maximum = 19.0)	• Nmap Quick Scan for hostname (19.0) • Device access with credentials (17.0) • Shodan search with organization name to find hostname (11.0) • Nmap Quick Scan for IP address (10.0)
Weighted Degree (Maximum =25)	• Google search for product Name (25) • Nmap Intense scan for all TCP ports (20) • Device access with credentials (19) • Nmap Quick Scan for hostname (19)

For the Weighted In-Degree measure, the top three nodes are searching for product, exploit data, and the Nmap Intense scan. This made sense to us because of key information that is needed to conduct these RTs: product name for product search, CVE or product info for the exploit search and hostname or IP address for the Nmap search. Many other RTs can provide these input data. For the Weighted Out-Degree, we saw that Nmap Quick Scans, Device Access, and Shodan search provide data that are used by a lot of other RTs. The Weighted Degree showed that the product name search, Nmap scans, and device access have a lot of "traffic" going through with "traffic" representing the total and in and out links. We use these RTs often in our reconnaissance work.

Table 4 shows values of the top-ranked RTs based on the Betweenness Centrality, Eigenvector Centrality, Authority, and Hub statistical results.

Table 4: Betweenness Centrality, Eigenvector Centrality, Authority and Hub Results for RT-Graph

Algorithmic Measure (Maximum Value for RT-Graph)	Top Results From RT-Graph (Measure Value)
Betweenness Centrality (Maximum = 1824.8)	• Google search for product name (1824.8) • Device access with credentials (1673.0) • Google search for default credentials (816.0) • Google search for technical manual with default credentials (816.0)
Eigenvector Centrality (Maximum = 1.0)	• ExploitDB search for exploits (1.0) • Google search for product (0.58) • NVD search (0.36) • Metasploit product search (0.36)
Authority (Maximum = 0.34)	• Nmap Intense scan for all TCP ports (0.34) • Nmap Intense scan with UDP ports (0.33) • GHDB for error messages, files with usernames, footholds, sensitive directories, vulnerable files, and vulnerable servers (all same value of 0.32)
Hub (Maximum = 0.50)	• Nmap Quick Scan for hostname (0.50) • Recon-ng Brute Hosts module for hostnames (0.42) • Recon-ng HackerTarget module for hostnames (0.40) • Shodan Search with organization name to see hostnames (0.35) • Google Search to find hostname (0.32)

The Betweenness Centrality measure shows the nodes in a graph that play a key role in linking parts of the graph together. The Google Search for Product Name RT and the Google Searches for Credential RTs (two of them) play a key role in linking Region 1 with Regions 2 and 3. Also, when a device is accessed it can be used to find a lot of other data such as access lists, logs, and more.

The Eigenvector Centrality measure shows the most critical nodes in a graph. ExploitDB holds a considerably higher value that other RTs. This made sense to us as the goal of reconnaissance is to support exploitation (MITRE, 2022). Additionally, NVD search and Metasploit product search also have higher values than most other nodes. These two also contribute considerably to supporting exploitation. And to use these two RTs, one needs to know the product name which is why the Google Search for Product RT is ranked high for Eigenvalue Centrality. We also saw that the that top four results are all for different tools and that the top three are all browser-based tools.

The Authority measure estimates the value of the content of the node. Nmap intense scans ranked at the top. They provide key data such as operating system (OS) version, service version, ports status and more. We see that six GHDB techniques also rank at the top. These techniques provide critical inside information across an organization beyond one device. It was interesting to see only two tools at the top of this measure and that 6 of the top 8 can all be used over the browser.

Finally, the Hub measure estimates the value of the links that nodes send to other nodes. It was interesting to see four different tools in the top five results. And it was interesting to see that each of these top-five RTs focused on finding hostnames. That shows how critical finding hostnames is in our reconnaissance body of knowledge.

5.4 Overall Comparison across Descriptive Metrics and Graph Analysis Algorithms

As we examined the results across the various data analysis techniques, we noted various trends. Overall, the RT of Google Search for Product Name represents a significant RT in our body of RT knowledge. It ranked 1 in the Betweenness Centrality, Weighted In-Degree, and Weighted Degree measures. It also ranked 2 in the Eigenvector measure. This shows how critical researching a product is in our reconnaissance approach.

After examining the top Eigenvector values in more detail, we found that 9/10 (90%) of the techniques rely on tools that can be used directly in the browser. Metasploit was the only one that could not be used directly in the browser. There are other ways to search for modules or exploits. however, our team uses Metasploit within Kali Linux.

We also see that Nmap is a powerful tool in our body of reconnaissance knowledge. Nmap as a tool that was associated with RTs that were ranked 1 for Weighed-In Degree, the Hub measure, and Authority. Also, Nmap was associated with the rank 2 and 4 for Weighed-In and Authority respectively. This made sense to us since we often use Nmap to find detailed information in reconnaissance.

We also looked at the graph data analysis results side-by-side with the descriptive metric results. It was interesting to note that ExploitDB is only used for one RT. However, this one RT became the most critical node in RT-Graph according to the Eigenvector Centrality metric. It was also interesting to know that GHDB is associated with six RTs and all six (100 percent) appeared as top-ranked Authority RTs. Finally, Metasploit is associated with two RTs but ranked fourth in the Eigenvector centrality metric.

One unexpected result was encountered. We often use Shodan in various RTs and value the information that we find from it. However, Shodan was not associated with many critical nodes based on graph statistical-algorithm results. We realized though we have used Shodan for 25 different RTs. We reasoned that perhaps because we use many Shodan RTs, it is difficult for one RT to stand out.

5.5 Future Work and Applications

The results of this research demonstrated which RTs our team relies on the most. It prompted us to explore other RTs and other reconnaissance tools such as Amass and SpiderFoot. Also, the research encouraged us to examine which RTs under ATT&CK are missing from our body of knowledge. Example ATT&CK RTs that we plan on exploring include Phishing for Information, Searching Closed Sources, and Searching Social Media. Furthermore, we plan on examining MITRE ATT&CK Tactics beyond Reconnaissance since Reconnaissance is only one of 14 tactics. Example tactics include Execution and Exfiltration. We also have formed a method for documenting new RTs and ingesting the new RT data into RT-Graph. We will use a shared Excel spreadsheet to track new tools, techniques, and syntaxes used. Then we will use the Protégé plugin named Cellfie to ingest the

Excel data into RT-Graph. Finally, as new knowledge is added to RT-Graph, our current RT-Graph can be compared to a future instance of our RT-Graph. Then the two RT-Graphs can be quantitatively and qualitatively compared to examine knowledge changes.

We see potential real-world applications of RT-Graph as our research continues to mature. For example, each individual team member brought different expertise across various RTs. Informally, various team members tested out new RTs that there weren't familiar with to expand their knowledge. In the future, we could examine how graph development or analyses increases one's knowledge. Also, we can search RT-graph by tool or by RT. The results can of the searches can serve as a checklist for completing reconnaissance projects for both experienced and new team members especially since each RT also comes with specific syntax to use.

6. Conclusion

ATT&CK provides a model of adversarial tactics and techniques including the Reconnaissance Tactic and its associated reconnaissance techniques (RTs). ATT&CK conceptualizes that RTs can inform other RTs. However, ATT&CK does not explicitly link RTs together. We created an ontology (Recontology) to model the Reconnaissance domain and then turned the ontology into a knowledge graph (RT-Graph) to explicitly link together 102 RTs. RT-Graph represents our body of knowledge in the reconnaissance domain. Using descriptive statistics, a graph spatial-layout algorithm, and graph statistical algorithms, we found the RTs that are most critical in RT-Graph, that have the most traffic going through them, that link main areas of RT-Graph together, that hold the most authoritative information, and the point to the most authoritative sources. Our research provides an approach for explicitly linking RTs together for using exploratory data analysis techniques to examine the RT knowledge graph. Future research will focus on building RT-Graph and comparing future instances of RT-Graph to the current one in order to examine knowledge growth.

Acknowledgements

The researchers acknowledge Addison Lilholt, from the Baldwin School, and Jeremy Slaven, from Drexel University, for their valuable feedback on our knowledge graph.

References

Aviad, A., Wecel, K. and Abramowicz, W. (2015) "The Semantic Approach to Cyber Security towards Ontology Based Body of Knowledge", *European Conference on Cyber Warfare and Security*.

Ding Z., Liu K., Liu B., and Zhu, X. (2020) "Survey of Cyber Security Knowledge Graph*", Journal of Huazhong University of Science and Technology,* Vol 49, No. 7, pp 79-91.

Dong, C., Jiang, B., Lu, Z., Liu, B., Li, N. and Ma, P. (2020) "Knowledge Graph for Cyberspace Security Intelligence: A Survey", *Journal of Cyber Security* Vol 5, pp 56–76.

Georgiadou, A., Mouzakitis, S., and Askounis, D. (2021) "Assessing MITRE ATT&CK Risk Using a Cyber-Security Culture Framework", *Sensors*, Vol 21, No. 9, pp 1-14.

Gruber, T.R. (1993) "A Translation Approach to Portable Ontology Specifications", *Knowledge Acquisition*, Vol 5, No. 2, pp 199-220.

Iannacone, M., Bohn, S., Nakamura, G., Gerth, J., Huffer, K., Bridges, R., Ferragut, E. and Goodall, J., "Developing an Ontology for Cyber Security Knowledge Graphs", *The 10th Annual Cyber and Information Security Research Conference*, pp 1-4.

Kurniawan, K., Ekelhart, A., Kiesling, E., Quirchmayr, G., and Tjoa, A. M. (2022). "KRYSTAL: Knowledge Graph-Based Framework for Tactical Attack Discovery in Audit Data" *Computers & Security*, Vol 121.

Langer, T., and Meisen, T. (2019) "Towards Utilizing Domain Expertise for Exploratory Data Analysis", *The 12th International Symposium on Visual Information Communication and Interaction,* pp 1-5.

Liu, K., Wang, F., Ding, Z., Liang, S., Yu, Z., and Zhou, Y. (2022) "Recent Progress of Using Knowledge Graph for Cybersecurity", *Electronics*, Vol 11*,* No. 15.

MITRE (2022) "Reconnaissance Tactic-TA0043", viewed January 1, 2023 <https://attack.mitre.org/tactics/TA0043/>

Noel, S., Harley, E., Tam, K.H., Limiero, M., and Share, M. (2016) "CyGraph: Graph-Based Analytics and Visualization for Cybersecurity." In *Handbook of Statistics*, Vol 35, pp 117–167.

Noel S. (2018) "A Review of Graph Approaches to Network Security Analytics", In *From Database to Cyber Security*, pp 300-323.

Noel, S., Harley, E., Tam, K. H., and Gyor, G. (2015) "Big-Data Architecture for Cyber Attack Graphs: Representing Security Relationships in NoSQL Graph Databases", *IEEE Symposium on Technologies for Homeland Security*.

Steinberg, A. N. (2017) "Foundations of Situation and Threat assessment", In *Handbook of Multisensor Data Fusion,* pp 457-522.

Sugumaran, V. and Storey, V. C. (2002) "An Ontology-Based Framework for Generating and Improving Database Design" *International Conference on Application of Natural Language to Information Systems,* pp. 1-12.

Syed Z, Padia A, Finin T, Matthews, L. and Joshi, A. (2016) "UCO: A Unified Cybersecurity Ontology"*, 30th AAAI conference on Artificial Intelligence*.

Yan Z. and Liu J. (2020) "A Review on Application of Knowledge Graph in Cybersecurity", *International Signal Processing, Communications and Engineering Management Conference (ISPCEM*), pp.240-243.

Ye, Z.W., Guo, Y.B., Li, T. and Ju, A.K. (2019) "Extended Attack Graph Generation Method Based on Knowledge Graph", *Computer Science*, Vol 46, No. 12, pp 165-173.

Zhang, K. and Liu, J. (2020) "Review on the Application of Knowledge Graph in Cyber Security Assessment" *IOP Conference Series: Materials Science and Engineering*, Vol. 768.