Global Information Assurance Certification Paper

Defending with Graphs:
Create a Graph Data Map to Visualize Pivot Paths

GIAC (GCIH) Gold Certification and RES-5500

Author: P. Brianne Fahey, thebriannefahey@gmail.com
Advisor: *David Hoelzer*

Accepted: 5/29/2019

Abstract

Preparations made during the Identify Function of the NIST Cybersecurity Framework can often pay dividends once an event response is warranted. Knowing what log data is available improves incident response readiness and providing a visual layout of those sources enables responders to pivot rapidly across relevant elements. Thinking in graphs is a multi-dimensional approach that improves upon defense that relies on one-dimensional lists and two-dimensional link analyses. This paper proposes a methodology to survey available data element relationships and apply a graph database schema to create a visual map. This graph data map can be used by analysts to query relationships and determine paths through the available data sources. A graph data map also allows for the consideration of log sources typically found in a SIEM alongside other data sources like an asset management database, application whitelist, or HR information which may be particularly useful for event context and to review potential Insider Threats. The templates and techniques described in this paper are available in GitHub for immediate use and further testing.

1. Introduction

Gathering intelligence and log evidence to support an investigation often requires having an intimate knowledge of the details that may be available across a vast array of log sets and data sources. The analysts' awareness of what log data is available and where it is stored increases readiness for incident response. Unfortunately, malicious actors rarely allow adequate time to abundantly prepare before launching an attack that a security analyst must quickly identify, isolate, and defend. Newer analysts do not have the luxury of years of training before jumping into incident response. At best, a less experienced analyst or a contract consultant brought in to respond to an incident will have a general idea of the system data in the environment and a framework to approach investigations to help overcome implicit bias (Sanders, 2016). For an analyst, having a map or guide to a new environment provides an advantage that turns a broad search for artifacts into a focused hunt.

To be able to think efficiently and pivot deftly through the environment to solve new challenges, an analyst should evolve from having a one-dimensional tool focus to having a two-dimensional link analysis and ultimately to multi-dimensional graph thinking. If an analyst can survey his or her environment, this information can be loaded into a graph database tool to provide a valuable visual map and a method to discover and quantify effective pivot routes through the data and systems to arrive at the desired output.

Cybersecurity defense and operations professionals must be prepared and trained to observe, orient, and act in response to any event. Attackers need only to exploit one weakness and lie in wait for an opportunity to pivot further along the kill chain and compromise an environment. To summarize, "Defenders think in lists. Attackers think in graphs. As long as this is true, attackers win" (Lambert, 2015).

Attackers often gain a foothold and pivot until they reach their goal. Defenders should use what they glean from adversarial tactics to meet the steep preparedness requirements. Analytic Pivoting, an analyst following links through data sets and systems in search of answers, is a critical skill for incident responders. It requires

extracting data, discovering related elements, forming a hypothesis, and testing it (Caltagirone, 2013). Analytic Pivoting is a practice made tangible with models, diagrams, and tools like a graph database. As an analyst gains more experience or more awareness of the unique technological environment in which he or she is working, the ability to pivot effectively is enhanced. An analyst's vision of the attack surface can morph into a multi-dimensional state that relies as much on intuition and familiarity as ability. Multi-dimensional thinking and simultaneous processing are more efficient than linear thinking.

1.1. Multi-Dimensional Pivot Evolution for Incident Response

Developing a multi-dimensional pivoting ability requires understanding individual data sources, capturing the links between them, and demonstrating holistic, interconnected relationships. A list is one-dimensional. A link analysis provides a two-dimensional view of related elements in the environment. A graph database depicts the multi-dimensional properties and relationships of many interconnected components.

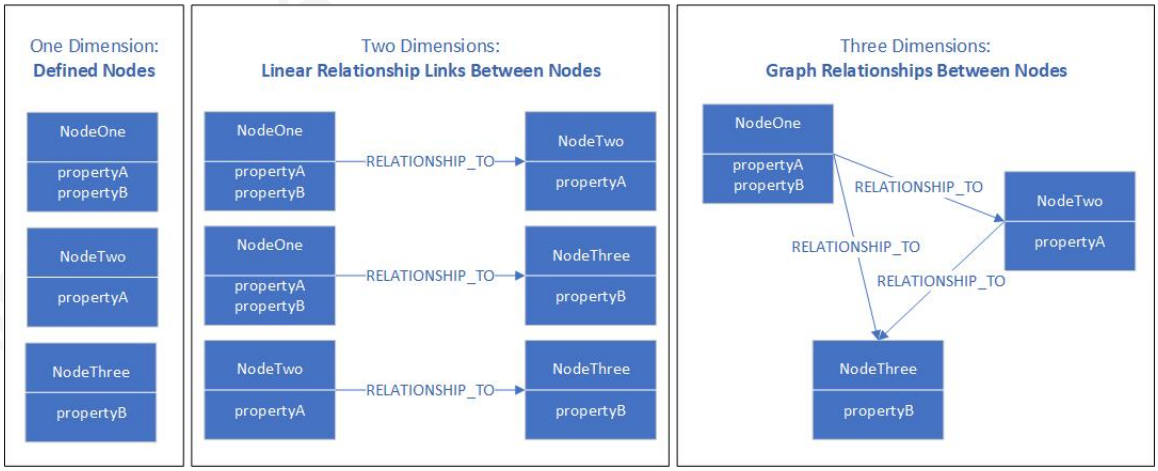


Figure 1. Maturing Dimensions of System Data Pivoting

1.1.1. Tactics

The hypothesis for this research is that analyst ability can be improved and the time it takes to investigate events can be reduced by creating a visual graph to examine links in available data sources. The research and data presented aim to establish a procedure to survey the system data in an environment, adopt a conventional schema to turn it into a graph database, and ultimately enable analysis gains based on the practical application of graph analysis. Creating a step-by-step model to achieve this provides a

tiered approach for analysts and engineers to survey, link, and develop a repeatable process to capture the current environment and onboard new system data over time.

2. Concept Review

Understanding the relationships between data is a significant tenet in both link analysis and graph analysis. Network theory describes link analysis as the connections between nodes. An example of this is an association matrix. Graph theory explores paths over nodes and edges. One of the most well-known illustrations of this is social network analysis.

2.1. Link Analysis

Link analysis is a vital step toward developing a sophisticated multi-dimensional graph data map. A link analysis explores relationships between entities; namely people or things. Link analysis techniques are used in law enforcement to demonstrate relationships between many actors and assets. Understanding the connections that a person of interest has to other people, places, or things may provide an investigation lead.

There are several promising applications of link analysis in cybersecurity; one such example is in creating and demonstrating links between available log and data sources to aid response preparedness and incident examination. This application helps to visualize outliers in a target-centric investigation and makes reports more compelling (Clemens, 2018). An example is the application of link analysis to OSINT through the use of a tool like Maltego.

Link analysis is comparable to enumerating or pivoting on a target node. If two systems are linked, an attacker can move from one to the other. It is vital for security defense operations to understand links in the environment. Link analysis of the connections for data elements surveyed could also be utilized to identify which data elements are the most available or to assess whether a tool will help fill a gap in the environment with the data elements it comprises.

2.2. Graph Analysis

Graph analysis is an aspect of Graph Theory. Graph Theory is the mathematical study of the structures built on relationships between points and lines. A graph is made up of vertices (or nodes) and edges (or links). Nodes are the nouns of the graph. These are objects given labels, described by properties, and connected to other nodes by relationship links. Relationships are the verbs of the graph and are the directional connections between nodes. Properties can also describe relationships. The image in Figure 2 below displays a visual graph of two account reps who work together named Bob and Cathy. Cathy, an award-winning account rep, hired Bob.

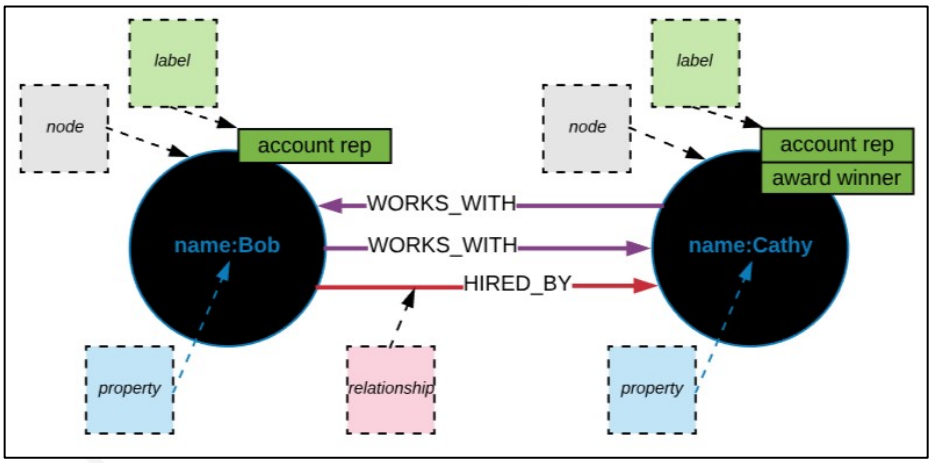


Figure 2. Modeling Nodes, Relationships, and Properties for a Graph

Node and relationship data populate a graph database which can generate images of the network of paths. Analysts can create a query to find a specific object or to learn about its relationships.

Many security professionals pursue excellence in pivoting skills for incident response. An illustration of this is the deep body of knowledge dedicated to Attack Graphs. An Attack Graph is a model to represent a scenario in which an offensive intrusion can be carried out. It can demonstrate potential vulnerabilities and exploits to the attack surface of an environment. The tool, BloodHound, (https://github.com/BloodHoundAD/BloodHound) uses graph theory to explore attack conditions with Active Directory.

3. Approach

The approach for this research is to design a method to survey available data sources and develop a tool to map collected information in a graph database. A survey of data elements available in security logs allows mapping for link analysis. Predetermined link analysis allows an investigating analyst to save time on speculative searching and instead follow the paths on a map to locate and extricate desired elements from specific logs. This graphical link analysis allows for the consideration of log sources typically found in a SIEM alongside other data sources like an asset management database, application whitelist, or HR information which may be particularly useful for event context and to review potential Insider Threats. A lab emulating a typical corporate environment including log source types such as flow data, firewall, web access logs, and other system logs will demonstrate this technique. A practical example of collecting log information needed to review a potential data exfiltration use case is performed before and after the link analysis is created to set a baseline and test the usefulness of the generated product.

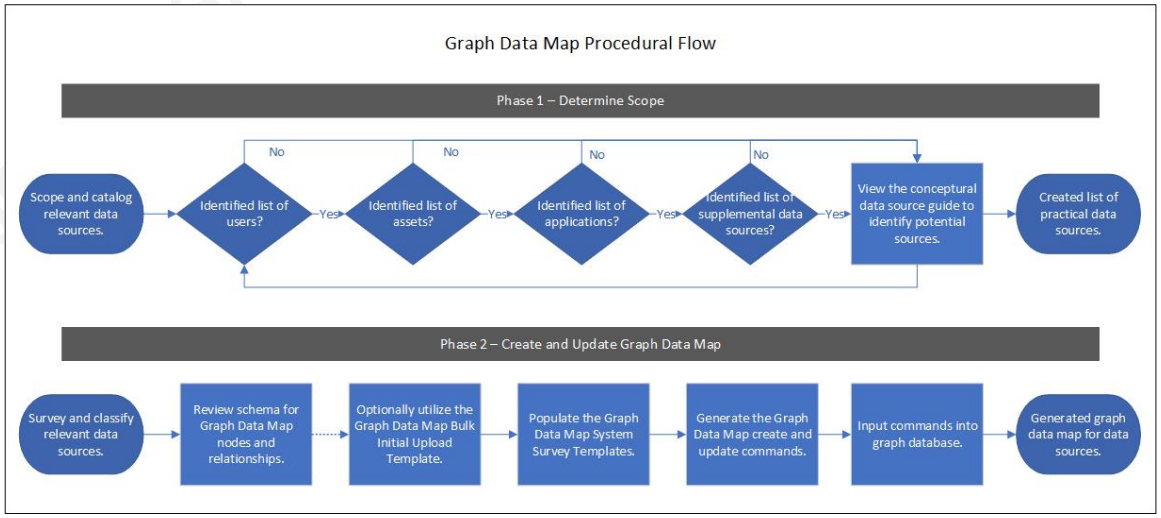


Figure 3. Flow Diagram of the Graph Data Map Procedure

3.1. Systems Data Scoping

A barrier to efficient incident response is analyst awareness of available data sources which are ever-changing. Decent notes on specific types of log collection within the environment could be hard to come by when an emergency or event arises. The goal

of systems data scoping is to generate an accurate list of the systems and devices in the environment.

3.1.1. Cataloging Relevant Data Sources

Creating a link analysis and pivotable defense map requires a thorough assessment of available data sources. The goal of this effort is to create an exhaustive list of available data sources that can be enumerated to elicit log relationships and attribute level details. Some of the work to determine all applications and assets in scope may be already complete. Many companies follow a security framework like NIST for IT Asset Management or a technology framework like ITIL for Asset and Configuration Management. Review the Configuration Management Database (CMDB) software library or application whitelist to sort through for relevant data sources. This dataset is a quality primary source because it references application owners and support resources to contact for further clarification. Review any available log management system or SIEM for an aggregated list of available data sources. Review the recent findings of any discovery or scanning tools. Scanned servers and workstations report back the applications and tools installed on these assets. Organizations should look beyond traditional security data sources to be able to fully consider threat intelligence and behavioral change indicators relevant to investigating insider threats.

A conceptual data source guide is provided in the Appendix [Figure A.1.] to suggest ideas of data sources to consider scoping into your graph data map. This activity can be broken up into a review of several categories. It is not meant to be a specific checklist to fulfill but rather it offers examples of free or open source tools that fit several common high-level categories or data types.

3.1.2. Practical Data Source Guide

The Conceptual Guide section above directed the evaluation of the lab scenario for potential data sources. This table contains the systems cataloged for the lab graph data map.

Data Category	Data Source Type	Data Source Name
Network and Web	NetFlow	AWS Virtual Private Cloud Flow Logs

IT Infrastructure	Asset or Config Management Database (CMDB)	List of Assets in the Environment
Email and Application	Web Applications	Apache Web Server Logs
Endpoint	Device Logging	Ubuntu Workstation & Server Syslog
Identity	Directory and Authentication	Ubuntu Workstation & Server Auth Logs
Context and Behavioral	Employee HR Profile Data	List of Active Employees
	Performance Sales Records	Employee Sales Trend Report
	OSINT and HUMINT	Call from Customer

Figure 4. Data Sources Scoped into the Virtual Lab Graph Data Map

3.2. Cataloging and Classifying Data Sources

To best analyze links in the data sources and meticulously populate a defense graph, the same data for each source identified in the data source table should be collected. This may require some exploration of the system log formats to understand what to capture. Adopting a standard format for a naming convention and classification schema ensures that as your collection of data sources grows, you can maintain the durable interconnected core goal of the data map. The first piece of building a graph data map is to set the schema and survey the systems to generate the graph database creation commands.

3.2.1. Schema for Nodes, Labels, Properties, and Relationships

The overall model for the graph data map schema features nodes, relationships, and properties as shown in the image below.

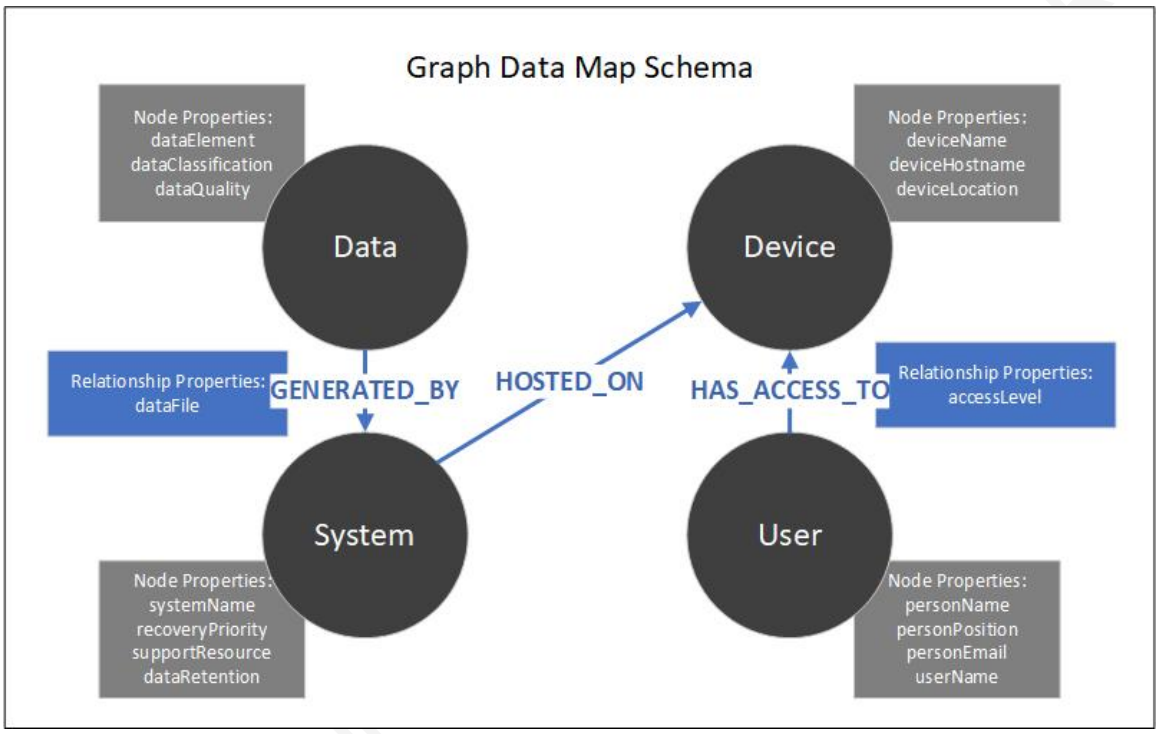


Figure 5. Model for Graph Data Map Schema

The schema uses four different node labels. Each represents a key type of node in the graph: System, Data, Device, and User. The table below displays samples using the Neo4j naming convention.

Node Label	Explanation	Sample Nodes	Sample Names
System	Denotes an application or tool.	awsFlow	AWS Flow Data
		apacheWebServer	Apache Web Server
		endpointWorkstation	Workstation Endpoint
Data	Denotes a file or element of a log file.	timestamp	Timestamp
		username	Username
		sourceIP	Source IP
Device	Denotes a server or asset.	cloudFlow	Corp NetFlow Cloud
		serverFileWeb	Corp File Web Server
		workstationOne	Corp Laptop One
User	Denotes a person.	josephineJones	Josephine Jones
		janetSamuel	Janet Samuel

Figure 6. Table of Graph Data Map Node Labels and Sample Names

P. Brianne Fahey, thebriannefahey@gmail.com

The schema uses three relationships. Each represents a directional link between two nodes. Figure 7 below displays samples using the Neo4j naming convention that can be read across like the example, "Data is generated by a system."

From	Relationship	To
Data	GENERATED_BY	System
System	HOSTED_ON	Device
User	HAS_ACCESS_TO	Device

Figure 7. Table of Graph Data Map Relationships

Combine the information in Figure 6 and Figure 7 to use the established naming convention and schema to find artifacts that answer the question: When is Josephine using her assigned computer?

- The user Josephine Jones has access to the device Corp Laptop One.
  josephineJones HAS_ACCESS_TO workstationOne
- The system Workstation Endpoint is hosted on the device Corp Laptop One.
  endpointWorkstation HOSTED_ON workstationOne
- Username and timestamp data is generated by the Workstation Endpoint system.
  username GENERATED_BY endpointWorkstation
  timestamp GENERATED_BY endpointWorkstation

Both nodes and relationships can have properties that describe and can be searched or correlated. Suggested properties for the nodes and relationships in the graph data map have been built into the schema but can be altered to fit the analyst requirements.

3.2.2. Populating System Survey Templates

A template workbook was developed to simplify the collection and conversion of the systems, devices, users, and data elements. This active workbook resides in the

GitHub repository https://github.com/theonlybrianne/graphdatamap. The image below shows how a completed system survey will look.

ONBOARD SYSTEM: AWS FIREWALL PROXY FLOW DATA	
1 - POPULATE SYSTEM SURVEY	
Define the System	
System Name	AWS Flow Data
System Recovery Expectation	Medium
System Support Resource	Josephine Jones
System Log Data Retention	7 Days
Define the System's Data Elements	
Log Data Element Name 1	Version
Log Data Element Name 2	Username
Log Data Element Name 3	InterfaceID
Log Data Element Name 4	Source IP
Log Data Element Name 5	Dest IP
Log Data Element Name 6	Source Port
Log Data Element Name 7	Dest Port
Log Data Element Name 8	Protocol
Log Data Element Name 9	Packets
Log Data Element Name 10	Bytes
Log Data Element Name 11	Timestamp
Define the System's Hosting Devices	
Host Device Name	Corp NetFlow Cloud
Host Device Hostname	cloudflow
Host Device Location	Cloud AWS
Define the System Admin User	
User Name	Josephine Jones
User Position	Technology Manager
User Email	admin@sellingtobuilders.com
User Computer Username	josephine
System Device Admin Level	root

Figure 8. Template - System Survey

3.2.3. Generating Graph Database Create and Update Commands

The same template workbook mentioned in the previous step contains procedural guidance and functions to help transform a system survey into a list of create and update commands that can be loaded into Neo4j to create your graph data map. This active workbook resides in the GitHub repository https://github.com/theonlybrianne/graphdatamap.

The system survey conversion is a bridge to the final goal of obtaining a set of create and update commands that can be input directly into the graph database. The template workbook contains functions that will generate the necessary commands based on the survey input and conversion stages. The image below shows what a completed set of create and update commands looks like within the template.

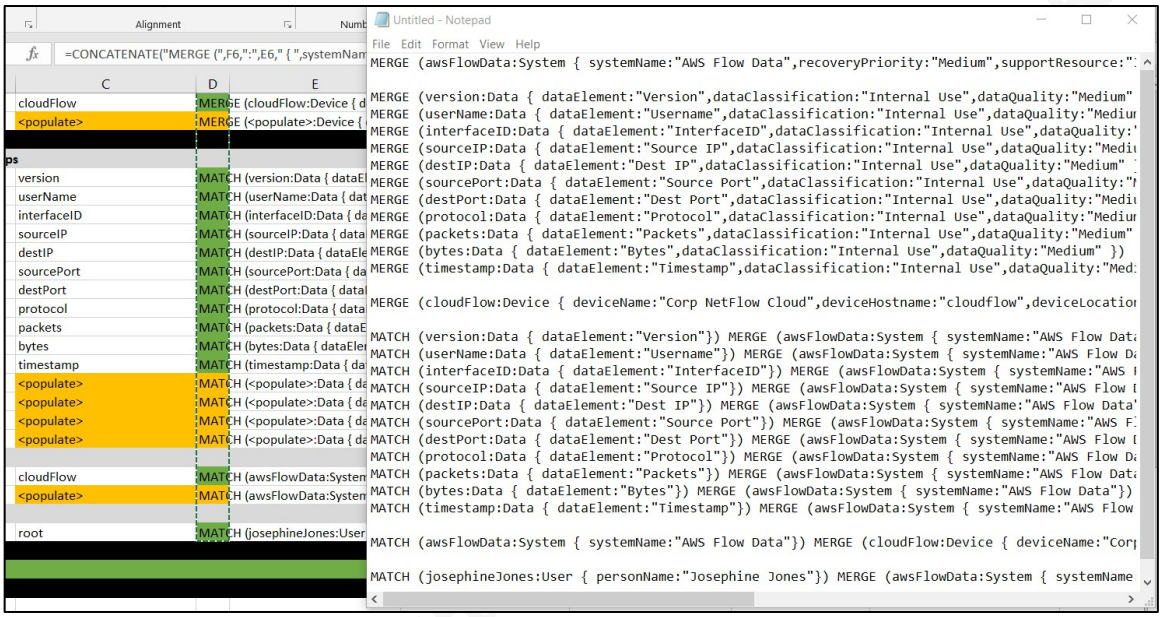


Figure 9. Template – Graph Database Create and Update Commands

3.3. Create a Data Map

The second piece of building a graph data map is to install an instance of Neo4j and load the create commands. Neo4j Community Edition is a graph database available for free. It and can be run via a desktop app and alternatively on a server hosted locally or in a cloud environment. The lab tested Neo4j Graph Database Community Edition installed from the AWS Marketplace in an Amazon EC2 instance. Once the software installs, it can be accessed through the browser GUI.

3.3.1. Getting Started with Neo4j

The completed graph data map system survey template provides the commands needed. Copy the commands into the Neo4j browser interface command line per the guidance in the instructions. Order matters. The template commands will first create the nodes with their labels and properties in one import. It may be cleanest to copy commands from the template workbook into notepad and ultimately into the Neo4j command window to allow for some visual input validation.

P. Brianne Fahey, thebriannefahey@gmail.com

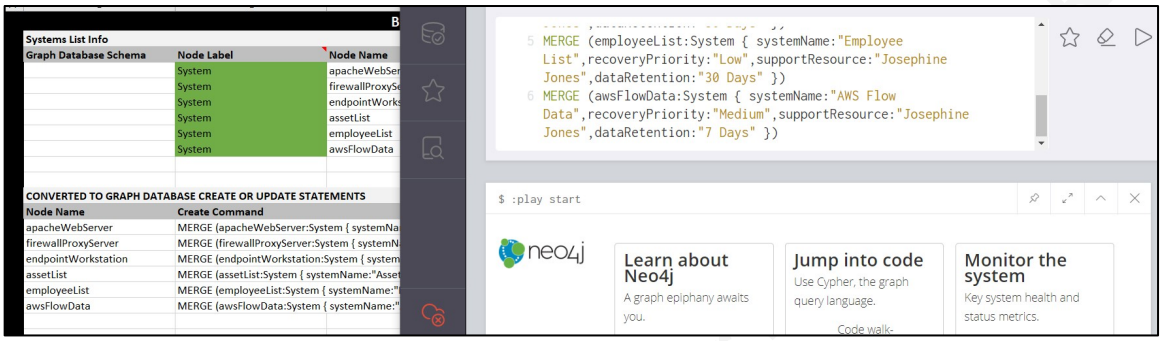


Figure 10. Copying Commands from Template to Neo4j Command Window

The template commands will then, one by one, establish relationships with properties between these nodes. Neo4j will respond in either a graph or table format depending on the input specified. The graph data map starts to take on an interconnected shape as analysts import more relationship commands.

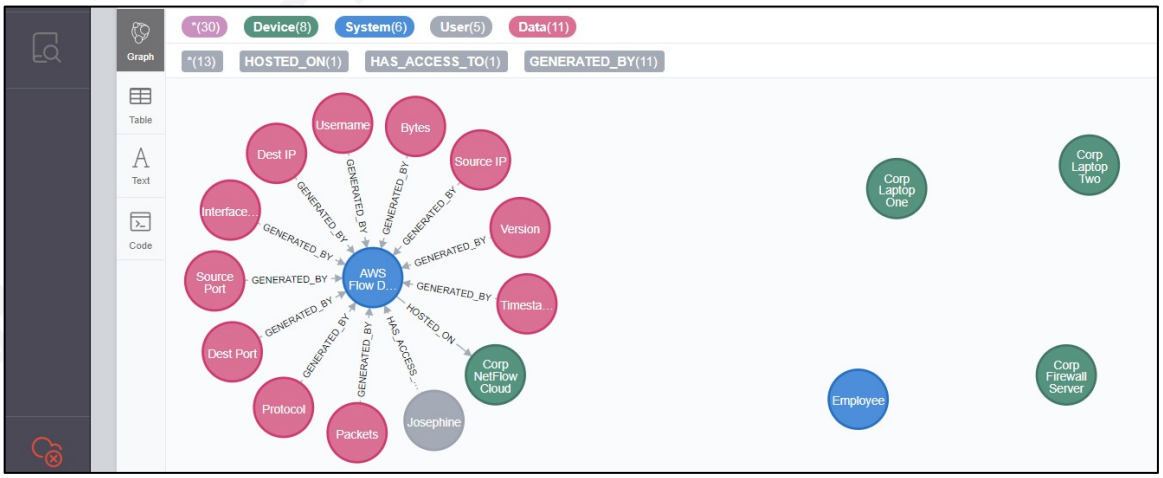


Figure 11. Neo4j Graph Data Map for an Onboarded System

Completing the import commands gives the analyst a complete graph data map to utilize. The image below is the full graph data map for the lab environment's use case testing.

P. Brianne Fahey, thebriannefahey@gmail.com

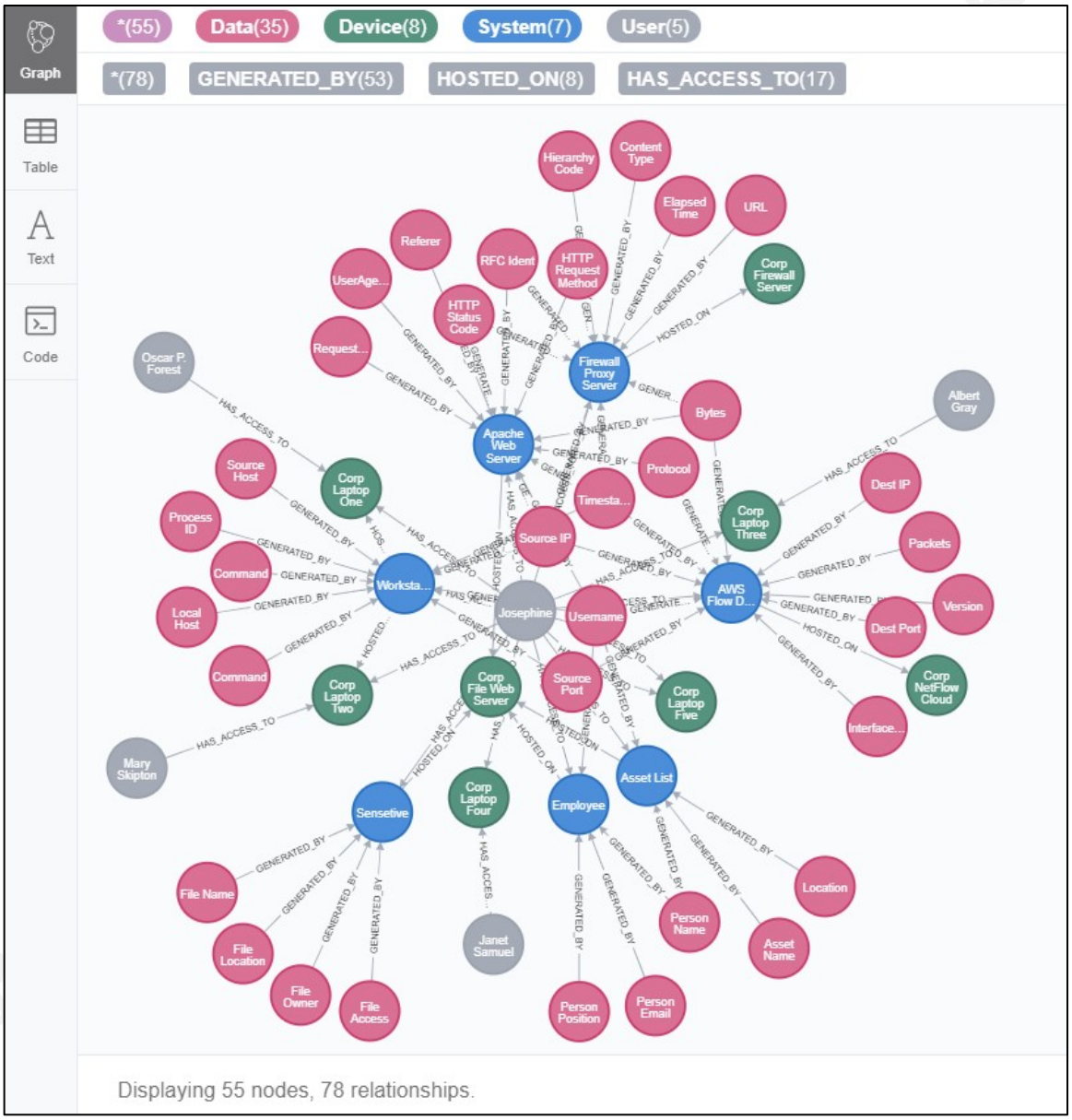


Figure 12. Full Graph Data Map for Virtual Lab Environment

3.3.2. Evolving with Data Source Changes

Analysts should expect constant change in the architecture of a corporate environment. Once the initial effort of database building is complete, the graph data map can accommodate growth and updates. Onboarding new items only require a survey and bulk node import. The generated commands are built to look for existing objects and update them or create new if it does not already exist. The Neo4j Cypher language allows detaching nodes from an established relationship and deleting them.

P. Brianne Fahey, thebriannefahey@gmail.com

3.4. Analyze Linked Paths in Data Map

The graph data map can be visually useful as the analyst drags and locks nodes on the interface to look at how they are connected. The primary benefit of the graph data map is the ability to query. The Neo4j Cypher query language enables an analyst to identify and extract specific nodes or sets of items that fit a specification. The query empowers an analyst to pin a location in the graph data map representing the company's data sources.

3.4.1. Wayfinding Linked Paths in the Graph Data Map

From the initial "you are here" pin, it is the job of the analyst to observe links and pivot to follow a path. The Neo4j Shortest Path algorithm provides a method to create the most direct line from the current position in the data to the desired location in the data. The query in the image below asks for the shortest path from a local host to a destination IP within the lab environment.

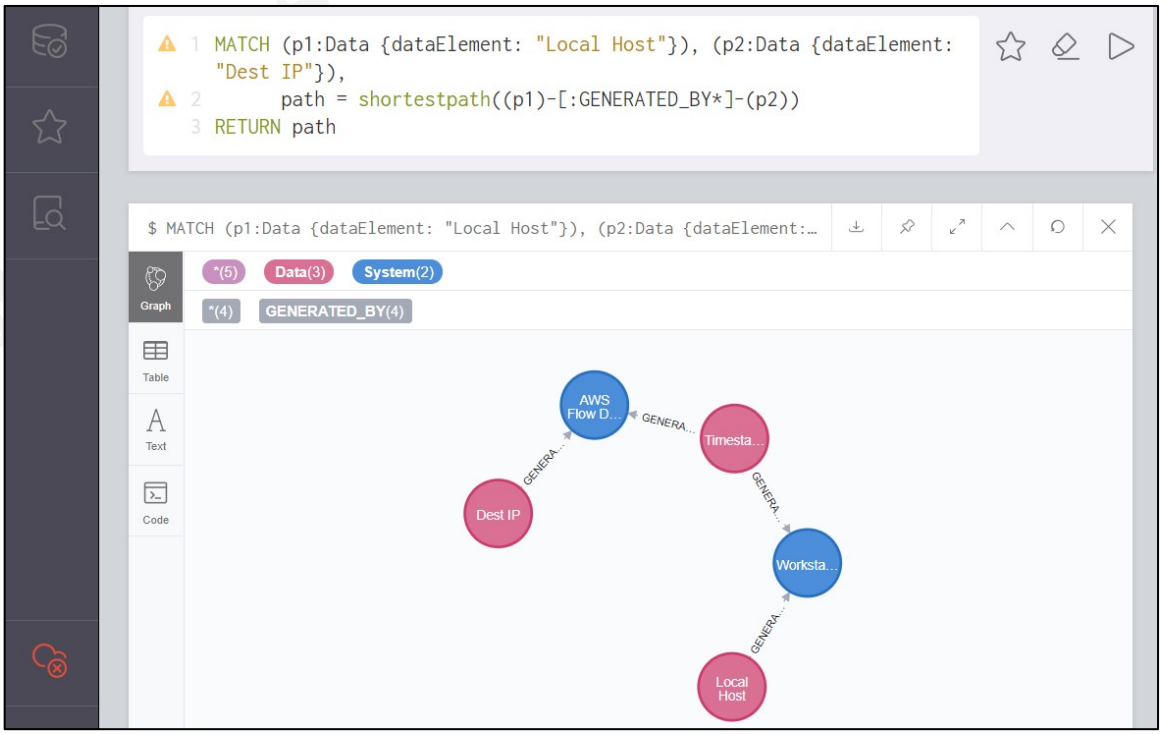


Figure 13. Graph Data Map Shortest Path Between Log Data Elements

4. Findings and Discussion

Nothing replaces earned experience and expertise, but tools can aid in the methodology learning curve and the organization of data. This analysis sets out to determine how to improve upon analyst pivoting and investigation; as such, it must start with a baseline activity.

A small business scenario is demonstrated in a virtual lab build. The environment represents the setup of a small construction sales firm called Sell to Builders. The firm is running open source office applications on low-cost AWS virtual servers. There are no security-specific platforms like an IDS or SIEM, but the network is secured with a firewall and proxy. The technology manager has tight user permissions set on the company web and file server and made sure that logging is available for all systems deployed in the company environment.

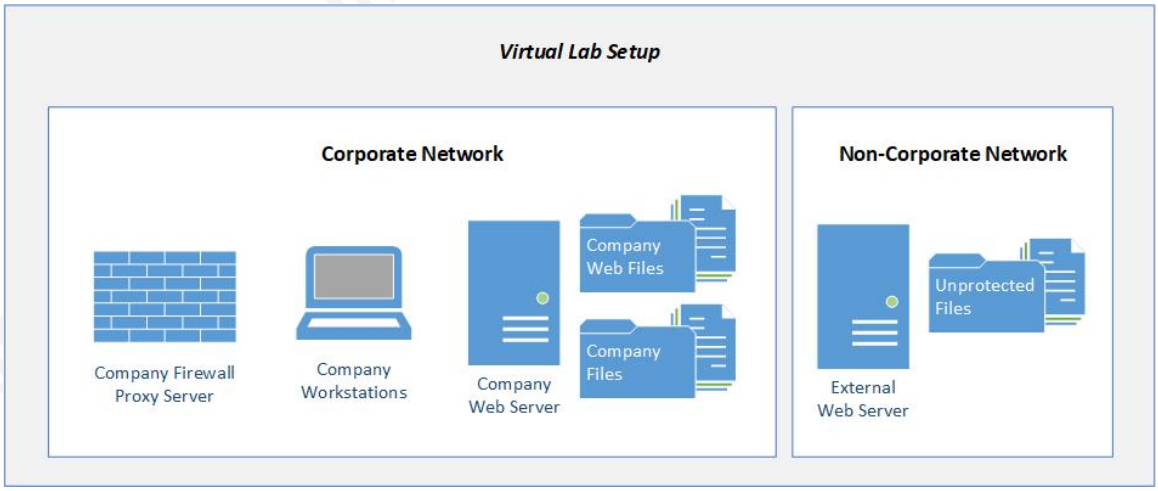


Figure 14. Representation of Virtual Lab Setup

The company owner has contracted a security analyst to review the event and determine whether there is a leak in the company. The company owner provided full access to the company systems and this information to the analyst:

A customer recently called to close his account at Sell to Builders. The customer mentioned that the competitor approached him and offered a deal he could not refuse. The competitor also told the customer he could work with a sales rep he knew from Sell

to Builders. Sell to Builders does keep a list of sensitive customer information on the file server, but permissions are limited to the company owner and the technology manager.

4.1. Scenario A: Incident Investigation without a Data Map

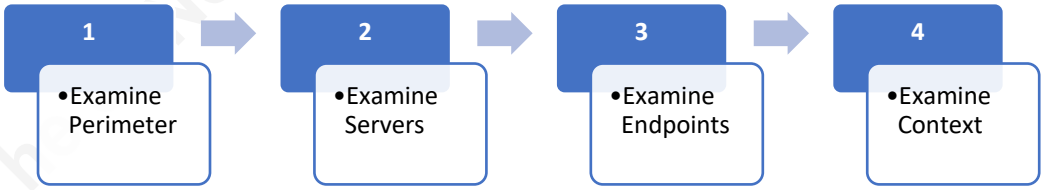
A virtual lab was created to demonstrate use cases. In this preliminary scenario, the use cases are executed in a lab environment with no graphical link analysis support. This scenario is considered a baseline investigation.

4.1.1. Analyst Investigation Approach

The analyst wants to answer these questions:

- Did the Sell to Builders customer list leave the network?
- If so, which user accessed and exfiltrated this sensitive data?

The analyst decides to check these logs to build a potential timeline of events:



1. Examine Perimeter: Review the Proxy, Firewall, and NetFlow Logs – Did the customer file leave the network?
    - Firewall and proxy logs look normal. Proxy logs show traffic in the network from the corporate web and file server to a corporate user workstation. NetFlow logs show communication with a workstation and an external IP that does not go through the proxy. A review of the external IP address linked it to a file sharing service.
2. Examine Servers: Review the File Server Access Logs – Who accessed the customer file? Review the File Server Auth Logs and Syslog– Who logged into the file server?

- The apache access logs on the web and file server show the user "josephine" accessing /files/CustomerData.xlsx. The file server syslog show "josephine" as the only corporate user who established a session. The file server auth logs show a failed SU for "opfor".

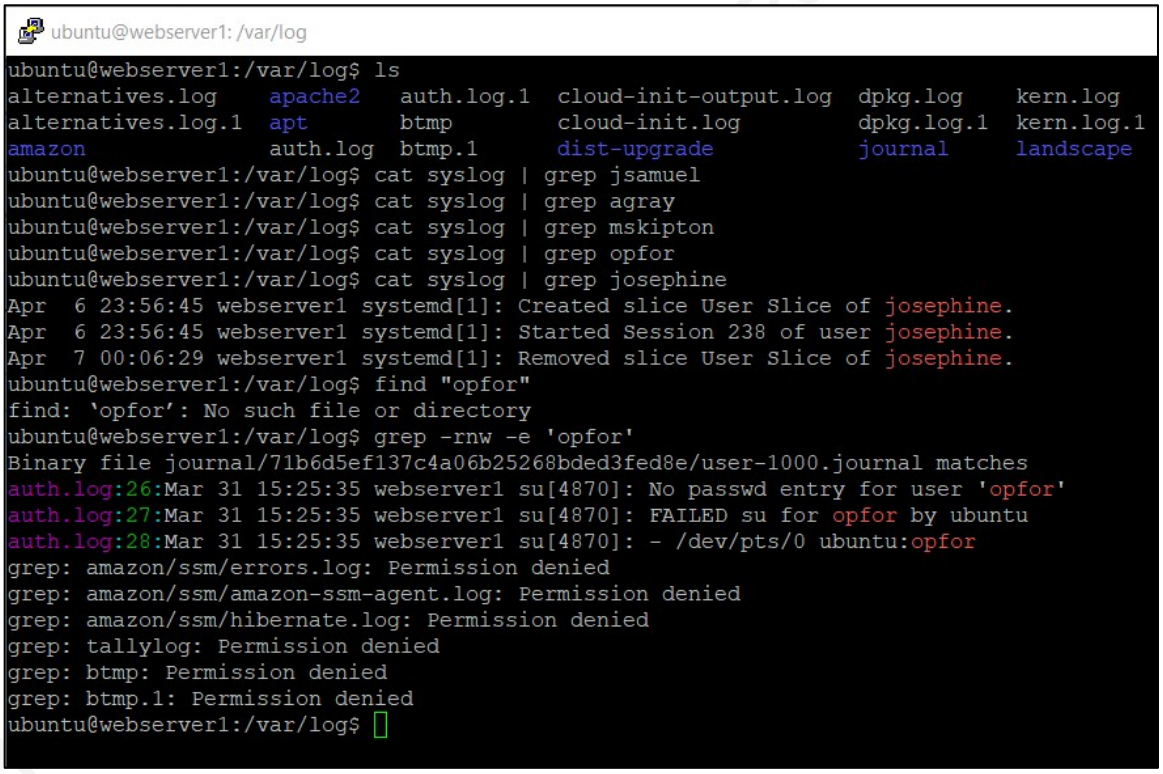


Figure 15. Corporate File Server Auth Log

2. Examine Endpoints: Review the "opfor" Workstation Access Logs – Who logged on to company workstations at the targeted time?

   - Workstation One auth logs show that "josephine" never logged into this workstation, only "opfor". Workstation bash history shows that "opfor" downloaded the customer data file, changed its name, and used SCP to send it off to an external file sharing service before deleting the file from his workstation. TestDisk tool shows the recent deletion of file MyData.xlsx from home/opfor.

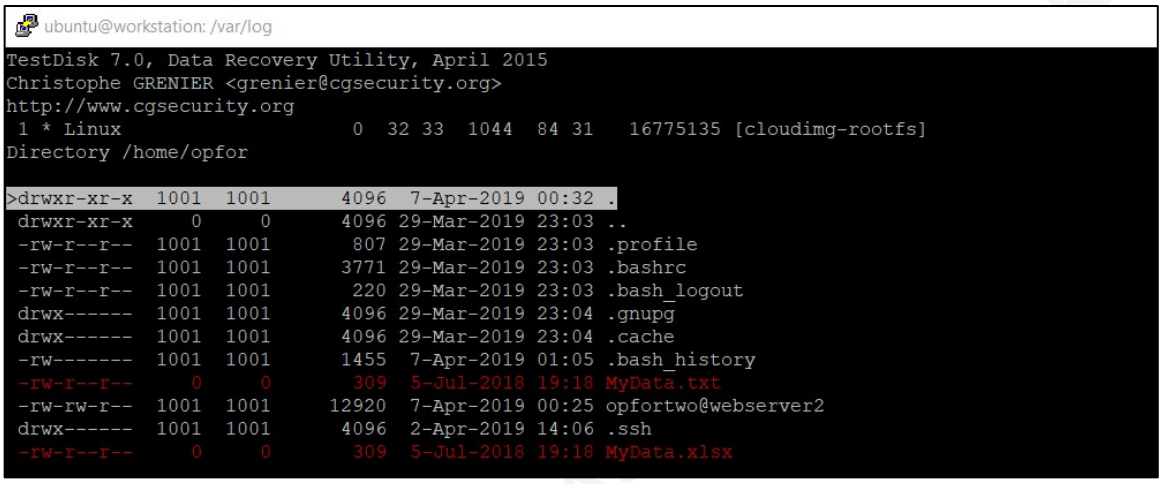


Figure 16. Evidence of Recently Deleted Files via TestDisk

3. Examine Context: Review the HR files, badge or camera data, or interview the users – What are the motives and whereabouts?

    - Analyst hypothesis is that user "opfor" utilized the admin account "josephine" to access the file server and download a customer file before sending it off to an external file sharing service. After interviewing two other colleagues Albert Gray and Mary Skipton, the analyst discovered that everyone knew Josephine Jones the technology manager has a terrible memory and tended to write down her passwords at her desk in the server room.

4.1.2. Summary of Investigation Efforts

To find the full timeline of events for this security breach, the analyst used his traditional method of working from the perimeter in toward the target endpoints, examining all available logs along the way.

This method took several hours and included filtering through days' worth of firewall, proxy, NetFlow, file server access logs, server auth logs, workstation auth logs, and ultimately some forensic activity on the endpoint targeted as the genesis of the event. The analyst also needed to gather context clues from the other employees and owner of the company to determine the reason why and explanation of how the event occurred.

4.2. Scenario B: Incident Investigation with a Data Map

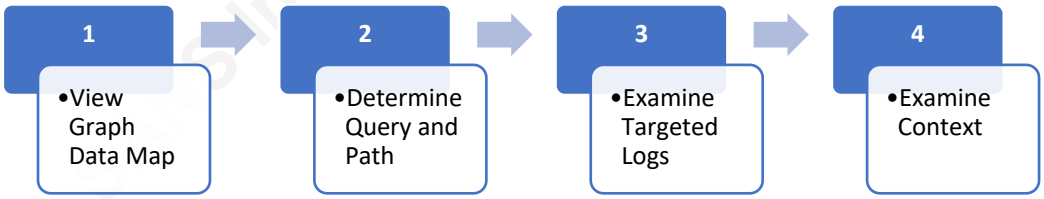
This scenario employs the same use case as the baseline investigation, but this time with the support of a graph data map for the analyst.

4.2.1. Analyst Investigation Approach

The analyst wants to answer the same questions:

- Did the Sell to Builders customer list leave the network?
- If so, which user accessed and exfiltrated this sensitive data?

The analyst decides to use the graph data map to view how the systems link together and determine what to query and on which path to focus before looking specifically into any log data sources.



1. View Graph Data Map: visually look at system connections and available log data elements in the context of the investigation.
    - The username "josephine" is connected to a high number of nodes. Properties in this node specify that Josephine Jones is the technology manager at Sell to Builders and root admin to all its systems. None of the other users have access to the Apache Web Server.

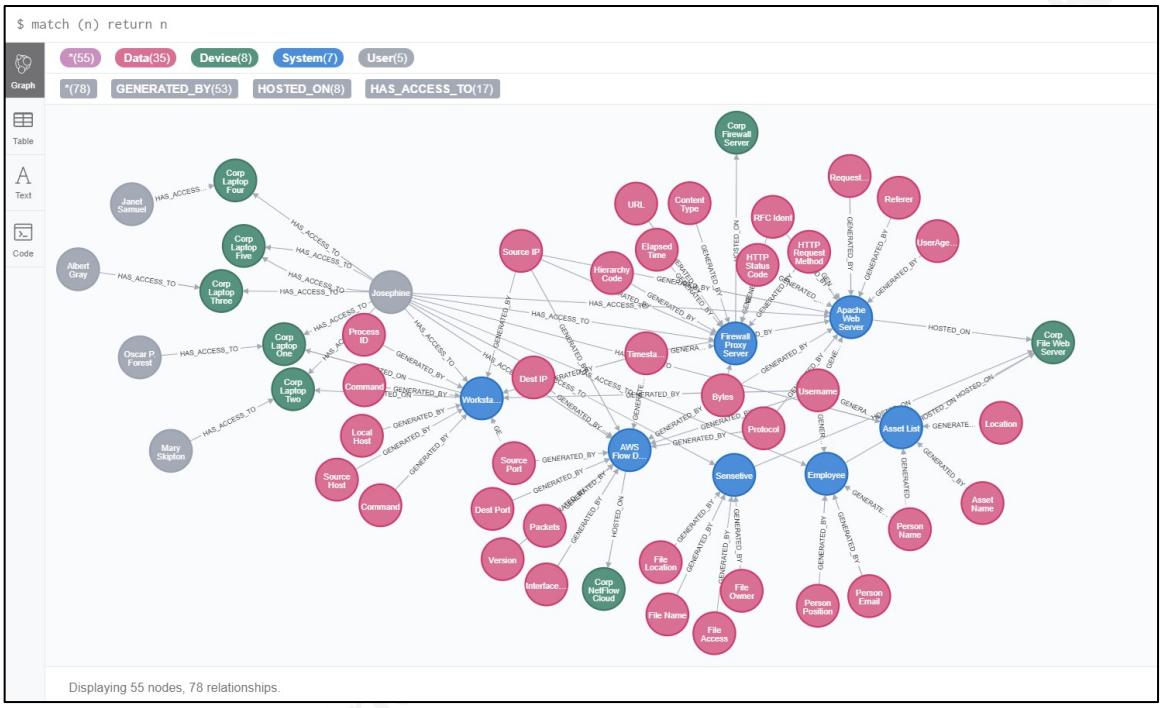


Figure 17. View of the Test Lab Graph Data Map

2. Determine Query and Path:

   - A destination IP would need to be linked with a local host or hostname in available logs if this were a data exfiltration event. The analyst queries the shortest path is between a log containing local host and one containing destination IP. The result directs the analyst to look at the AWS flow logs and the workstation logs.

3. Examine Targeted Logs:

   - AWS Flow logs show traffic to a destination IP which is known to be an external file sharing service and did not go through the proxy. That traffic was generated by workstation one which the graph data map properties explain is used by Oscar Forest (username "opfor") and administered by Josephine (username "josephine"). Josephine's workstation logs show that she was not logged in to her laptop during

the time of that suspicious traffic. Oscar's workstation logs show that he was logged in at that time.

- Oscar's bash history shows that he failed an attempt to su to the file server and then successfully accessed a file. By adjusting the query to see how the username "opfor" connects to the file server, it is clear that "josephine" is necessary on that path.

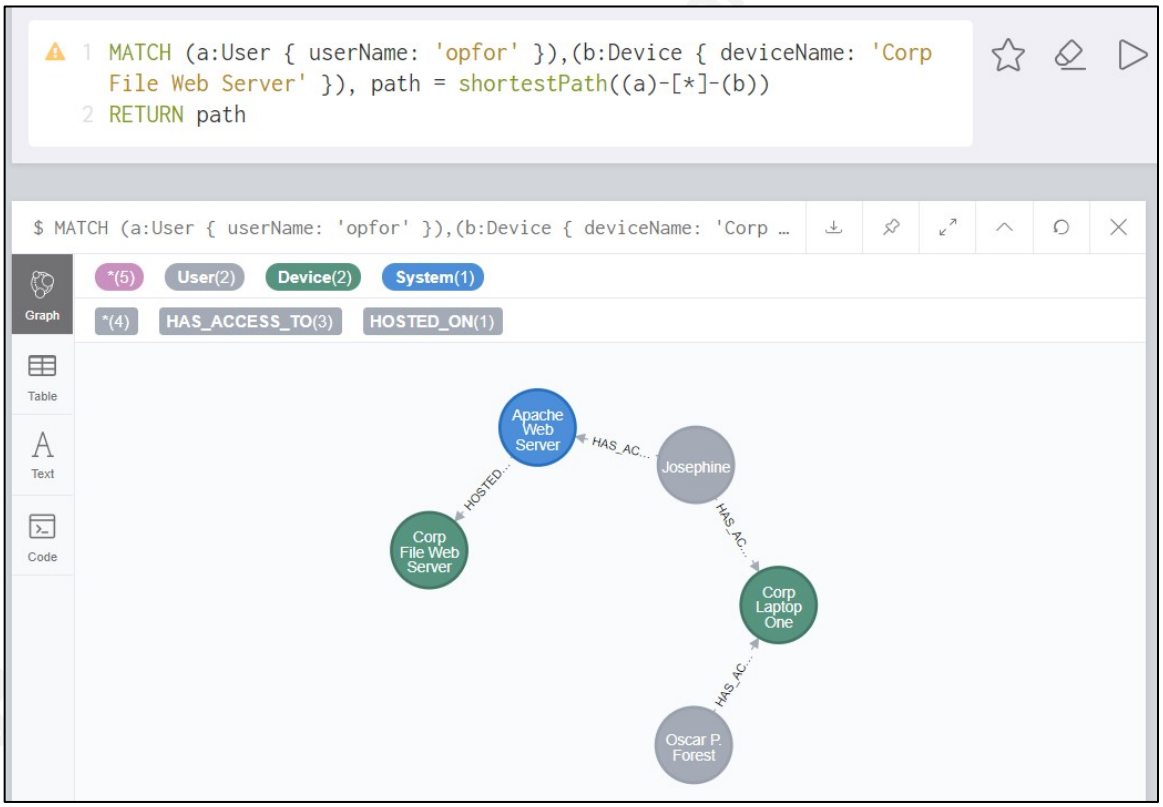


Figure 18. Data Map Shortest Path from Oscar's Workstation to File Server

4. Examine Context: interview the employees under the new hypothesis.

   - The analyst hypothesis is that "opfor" logged into the corporate file server with Josephine's credentials to download and then exfiltrate a customer file. Interviews with the owner Janet Samuel and other sales representatives validate the hypothesis.

4.2.2. Summary of Investigation Efforts

To find the full timeline of events for this security breach, the analyst used the graph data map to focus in on which systems and users were crucial to data exfiltration. The user formed a hypothesis about a potential data exfiltration scenario executed by Josephine before spending time searching through any log sources. The analyst was able to lean on the graph data map to re-frame and pivot on a new path when the logs disproved the initial hypothesis.

This method took a few hours and only required filtering through days' worth of NetFlow before a finding refocused the analyst on a new linked path. The analyst was able to look at the big picture and then perform more focused, iterative searches to find a first-rate hypothesis that could be used to gather information in an interview.

4.3. Detailed Use Case Summary

The use case performed is a small business insider threat case where a disgruntled user escalated privilege to steal a customer file and provide it to a competitor. A sales rep named Oscar Forest recently left his job at Sell to Builders after he received no annual bonus after several quarters of declining sales. Before he left, Oscar attempted to use his account to access the Customer Data list on the file server. When he discovered he could not access the file, he found the password for the admin account "josephine" and used it to access and download the Customer Data file. Oscar downloaded the Customer Data file to his desktop and changed its name before transferring it off to a file sharing service outside the corporate network and deleting it from his corporate workstation.

4.4. Comparison of Investigations with and without Data Map

Utilizing a graph data map in an investigation is less comfortable because it is unfamiliar. However, the value of seeing the shape of the environment and even querying a data element to create a path linking data sets is clear. The best way to compare the baseline use case investigation with the graph data map use case investigation is to discuss the time, tools, and talent needed to create and utilize a graph data map.

Factor	Graph Data Map Proposal
Tools	• Build: System survey template workbook available on GitHub. Neo4j graph database software community edition is free. • Scale: As node and relationship volume increases and queries become intricate, the system could need higher capacity processing and storage.
Time	• Build: Approximate several days' work to scope and survey data sources as well as create graph data map. • Maintain: Occasional time for onboarding new sources and updating current sources may be needed ongoing. • Investigate: Experience a learning curve with the introduction of this new method. Long term goal is to save time by increasing the ability to focus an analysis expedition through data sources.
Talent	• Build: Technical skills needed include using a template workbook and installing Neo4j software. • Investigate: Analysts benefit from learning pieces of Cypher query language to build paths and optimize searches through a graph database.

Figure 19. Table Comparing Time, Talent, and Tools

A graph data map will be valuable to an organization that has a mixture of junior and senior analysts. The more experienced analysts will be in a position to consult on which data source need to be scoped in and surveyed for the graph data map. The less experienced analysts or technical business analysts may have the time between incidents to put in the initial time to build the graph data map. There is virtually no cost associated with building a graph data map. Neo4j offers a free community version. The developed templates are available on GitHub. There is also not a high technical barrier to entry with the process utilizing a workbook

P. Brianne Fahey, thebriannefahey@gmail.com

5. Recommendations and Implications

A graph data map for cyber defense investigations has a high probability of coming into use and even being built into current security platforms like the SIEM. A key differentiator is the ability to take advantage of use cases that will benefit both the business and the security team.

5.1. Recommendations for Practice (or for Use in the Field)

Visuals generated by a graph data map are sensitive and should only be used internally to the company. In organizations where there is a Security Operations Center facility featuring display monitors, there is an opportunity to include the graph data map among those screens. Analysts can use the visual to discuss tactics and approaches to determining the cause of an event or even to plan potential threat hunts.

If the visual could simplify to a higher-level node schema, a graph data map could play prominently in a tabletop exercise involving parts of the business outside of security like compliance and operational resources. The graph data map acts as a picture to help explain the state to a business partner with expertise in finance or law instead of networked computer systems. Visual learners will have a map in front of them to reference where the current concern lies.

An advantage of having a graph data map outside of a SIEM is that it encourages the consideration of data sources outside the standard. Integrating non-traditional security sources like HR data and physical locations with security sources will enable Insider Threat investigations like information technology sabotage or even fraud.

5.2. Implications for Future Research

The best path forward for a graph data map is to combine with complementary development projects within the information security and data analysis communities. Some examples of synergies include:

- Chris Sanders' pivotmap tool (https://github.com/chrissanders/pivotmap) also explores creating a pivot map for analysis data sources. Additionally, his Investigation

Theory course at Applied Network Defense delves into assessing the quality of data sources. This work is a natural fit in conjunction with weighted path queries in the graph data map to calculate not only the shortest path but also the route through the most reliable data sources.

- Colin O'Brien's grapl platform (https://github.com/insanitybit/grapl) utilizes graph analysis on security detection and response. This work goes beyond the surface relational link between sources and turns raw logs into graphs to develop signatures for review.
- Olaf Hartong's ATTACK datamap tool (https://github.com/olafhartong/ATTACKdatamap) aligns data sources to the MITRE ATT&CK framework categories. This project demonstrates another approach to determine the best data sources for the situation.

6. Conclusion

In an environment driven by vast amounts of data, any method to help focus an analyst's search is valuable. Link and graph analysis are valid approaches to cybersecurity incident investigation and response due to their ability to enumerate pivot points in complex log aggregations.

The graph data map described in this paper is a low-cost, low-tech way to create a graph database of the available data sources in the environment. The template materials are available to download on GitHub for use and collaborative improvements. This tool builds on the experienced law enforcement methodology of link analysis. It enables and encourages the mindset of defenders taking a page out of the attackers' book and thinking in graphs versus lists. Graph analysis complements the idea of defense in depth. The graph data map can mature by looking for automation opportunities to minimize ongoing maintenance time. It can strengthen by further developing the natural overlaps with existing analyst theory and graph analysis work in the cybersecurity space.

P. Brianne Fahey, thebriannefahey@gmail.com

References

Brath, R., & Jonker, D. (2015). *Graph Analysis and Visualization: Discovering Business Opportunity in Linked Data*. Hoboken, NJ: John Wiley & Sons.

Caltagirone, S., Pendergast, A., & Betz, C. (2013). *The diamond model of intrusion analysis*. Retrieved from http://www.activeresponse.org/wp-content/uploads/2013/07/diamond.pdf

Clemens, D. (2018, June 21). Understanding link analysis and using it in investigations - ShadowDragon [Web log post]. Retrieved from https://shadowdragon.io/blog/understanding-link-analysis-and-using-it-in-investigations/

Criminal intelligence manual for analysts. (2011). Retrieved from https://www.unodc.org/documents/organized-crime/Law-Enforcement/Criminal_Intelligence_for_Analysts.pdf

Fahey, P. B. (2019, April 12). theonlybrianne/graphdatamap. Retrieved April 13, 2019, from https://github.com/theonlybrianne/graphdatamap

Hartong, O. (2019, April 7). Assess your data potential with ATT&CK Datamap [Web log post]. Retrieved from https://medium.com/@olafhartong/assess-your-data-potential-with-att-ck-datamap-f44884cfed11

Lambert, J. (2015, April 26). Defenders think in lists. Attackers think in graphs. As long as this is true, attackers win [Web log post]. Retrieved from https://github.com/JohnLaTwC/Shared/blob/master/Defenders%20think%20in%20lists.%20Attackers%20think%20in%20graphs.%20As%20long%20as%20this%20is%20true%2C%20attackers%20win.md

P. Brianne Fahey, thebriannefahey@gmail.com

Neo4j Inc. (2019). The Neo4j operations manual v3.5 [Web log post]. Retrieved from https://neo4j.com/docs/operations-manual/3.5/

Niese, P. (2016). *Intrusion detection through relationship analysis*. Retrieved from SANS Technology Institute website: https://www.sans.org/reading-room/whitepapers/detection/intrusion-detection-relationship-analysis-37352

O'Brien, C. (2019, April 8). insanitybit/grapl. Retrieved April 13, 2019, from https://github.com/insanitybit/grapl

Robbins, A., Vazarkar, R., & Schroeder, W. (2019, April 3). BloodHoundAD/BloodHound. Retrieved from https://github.com/BloodHoundAD/BloodHound

Sanders, C. (2016, May 4). How analysts approach investigations [Web log post]. Retrieved from https://chrissanders.org/2016/05/how-analysts-approach-investigations/

Sanders, C. (2017, September 11). chrissanders/pivotmap. Retrieved from https://github.com/chrissanders/pivotmap

Sanders, C. (2016). *Investigation theory the analyst mindset: Grading data sources*. Retrieved from https://chrissanders.org/training/investigationtheory/

P. Brianne Fahey, thebriannefahey@gmail.com

Appendix

A.1. Systems Data Scoping: Conceptual Data Source Guide

This activity can be broken up into a review of several categories. This is not meant to be a comprehensive checklist to fulfill. It is providing several common high-level categories, types of data Included, and free or open sources examples of tools.

Data Category	Data Source Type	Free or Open Source Examples
Security Correlation	Security Incident Event Management (SIEM)	Splunk Free, OSSIM, OSSEC
	Intrusion Detection System (IDS)	Snort, Suricata, Zeek, Security Onion
	Log Management System (LMS)	Graylog, ELK Stack, Fluentd, syslog-ng
	Vulnerability Management (VM)	Nessus Free, OpenVAS, Nexpose Community
Network and Web	NetFlow	SiLK, Ntop, PRTG Free, Flowscan
	Firewall	pfSense, IPFire, NG Firewall
	Proxy	Squid, HAProxy, Nginx
	IP Address Management (IPAM)	phpIPAM, NIPAP, netbox
Database and Storage	File Sharing and Integrity Monitoring (FIM)	SeaFile, FreeNAS, Netwrix, Tripwire Open Source, OSSEC, Wazuh
	Database Management and Activity Monitoring (DBMS, DAM)	Nagios, sqlShark, OmniDB
IT Infrastructure	Asset Management, Config Management Database (CMDB)	GLPI, SysAid, Open-AudIT
	Software Library, Application Whitelist	Spiceworks, OSSIM, SQLite
	Scanning and Discovery Tools	Nmap, OpenNMS, ipscan, masscan
Email and Application	Web Applications	Apache, Nginx, Lighttpd, Payara
	Email	Zimbra, Kolab, Apache James, Postfix
Endpoint	Endpoint Detection and Response (EDR)	Wazuh, Comodo cWatch EDR
	Device Logging	Syslog, sysmon, auth logs, access logs
Identity	Directory and Authentication	OpenLDAP, ApacheDS, OpenDJ

P. Brianne Fahey, thebriannefahey@gmail.com

	Identity Management, Account Ownership	OpenIAM, Apache Syncope, Gluu
Information Classification	Sensitive Data	Imperva, weka
Physical Security	Location and Badge Data	Envoy basic, identiv uTrust TS Cards
Context and Behavioral	Threat Intelligence Feeds	ThreatConnect Open, MISP Threat Sharing
	Employee HR Profile Data	Factorial, zenefits
	Performance and Sales Records	Bitrix24, zoho crm
	Contractor or Vendor Management Data	Agiloft Free
	Location, Time Tracking	OpenProject, Monday, Google Apps
	OSINT and HUMINT	Maltego, recon-ng, Buscador

Figure A1. Conceptual Data Source Guide