Graph Analytics Based Security Framework

Rupesh Shantamurty, Sridhar Bandi, Logeswari P Viswanath, Vikrant Mah Dhimate
Compute Software Engineering
rupeshs@hpe.com, sridhar.bandi@hpe.com, logeswari.pv@hpe.com, vikrant.mah.dhimate@hpe.com

Abstract

Managing the security of IT infrastructure is an ever evolving field. What matters is adapting quickly to the technological developments. Now a days in IT infrastructure deployments the security controls are very generic in nature due to lack of insights into the workloads deployed in those infrastructure. Even if there is awareness about the workloads running on certain machines the security controls are implemented manually. Due to this the attack vector is quite large and if one of the machines gets compromised then the whole infrastructure is under threat. Here in this paper we solve this problem by proposing an innovative intelligent framework which utilizes the analytics capabilities of graph mining on the data relating to infrastructure to build custom security controls around critical components of the infrastructure. By this not only do we reduce the attack vector we also implement the core security principle of defense in depth so that the impact of the compromised machine is limited.

Problem statement

In the complex data center with tens of thousands of servers it's difficult to identify machines running critical workloads.

This usually happens because of the combined effect of the below factors :

- Adoption of Virtualizations & Containerization technologies
- Distributed development and adoption of DevOps model

Since it's difficult to identify critical infrastructure components the security model and controls are more generalized and this increases the risk exposure when a breach happens.

Our solution

We present here an automated framework which utilizes graph mining techniques to identify the critical components in a complex mesh of IT infrastructure. By identifying the critical infrastructures components we can then tailor the security controls around them either in automated manner or manually.

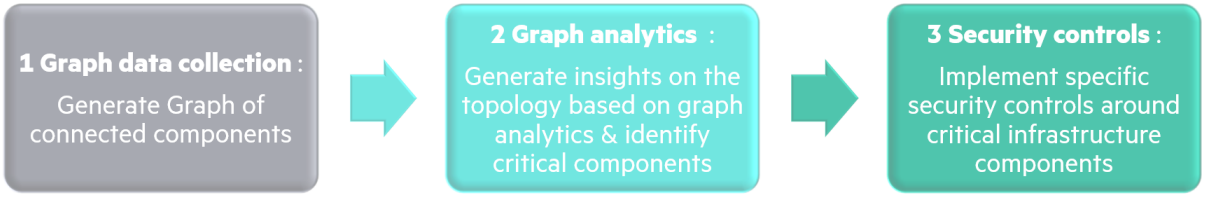

Figure 1: Workflow of Graph Analytics Based Security Framework

Our solution has three modules as stated below :

1. **Graph data collection module :** In this module we collect the data about the nodes in the IT infrastructure and how they interact with each other. By this we can identify which nodes connect to which other nodes. For physical machines this data is collected by capturing network topology data using tools like NeDi[1] to generate the topology graph identifying the nodes and connections. The data relating to where the containers and web services are deployed and how they interact with each other is obtained from the service mesh information.

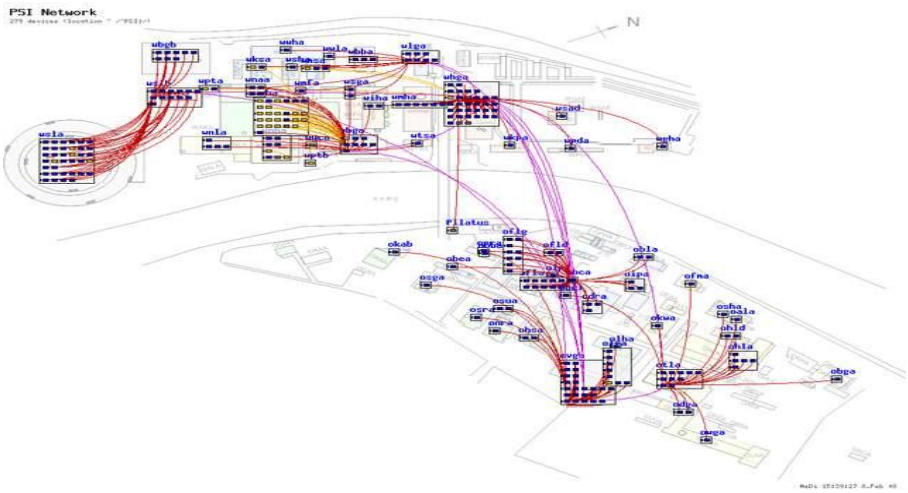

Figure 2: Graph data generated by NeDi

2. **Graph analytics module :** In this module we use graph analytic techniques on the collected graph data to identify the critical components

    a. Identifying communities using graph clustering/partitioning algorithms, like k_clusterSpanningTree, SNN_Clustering(SNN)

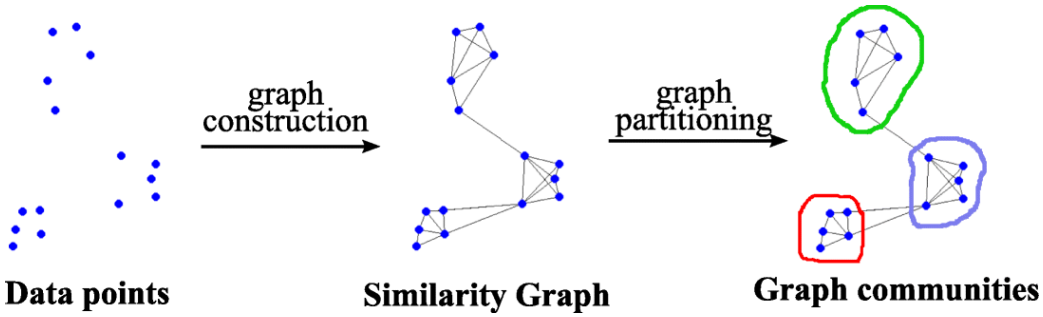

Figure 3: Schematic diagram of the workflow for graph-based clustering [2]

    b. Identify critical components as most connected nodes using centrality identification mechanisms. A centrality measure may be regarded as a measure of importance of a particular entity. Various centrality measures exist but we have currently considered only degree centrality.

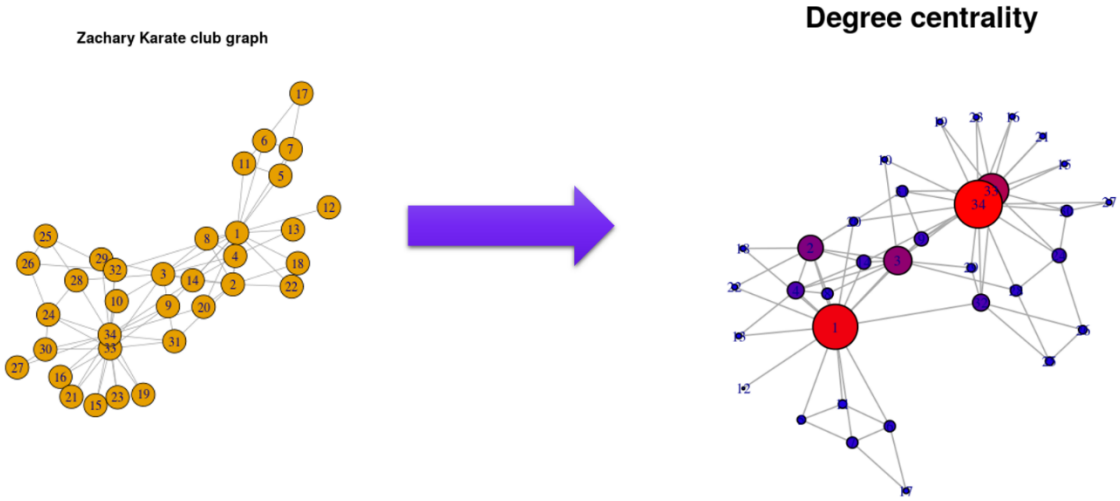

Figure 4: Compute degree centrality to identify the most connected nodes, here 1 and 34

3. **Security controls module:** The default security configuration of most operating environments and servers might not be enough to provide enough protection to critical components. Hence, once the communities and critical nodes are identified then in this module we customize the security controls around those communities and the critical nodes.
    a. Firewall around communities identified in the graph

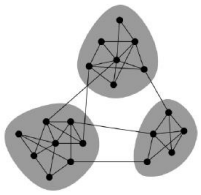

Figure 5: Firewalls around communities of nodes

    b. Security hardening for critical nodes
        i. Operating system hardening
        ii. Workload aware hardening

Evidence the solution works

Each of the modules mentioned above are known to be working independently. Here in this framework we are working towards stitching them together to solve the problem as stated in the problem statement.

Competitive approaches

Firewalls have evolved over the ages to be more intelligent but they are meant to protect the security perimeter and do not present a fluid framework as presented in this paper. Considering cloud infrastructure, the CASB[3] solutions also do no provide a holistic approach as presented in this paper.

Current status

Currently we are working on the proof of concept (POC) using data collected using NeDi. For graph analytics we are using python and SNAP[4], which contains parallel implementations of fundamental graph-theoretic kernels and optimized implementations for search, connected components, vertex and edge centrality. We are applying the security controls on multi node infrastructure hosting web services based application deployed using kubernetes containers. We can demonstrate the POC by the time of techcon.

Next steps

Once the POC is ready we plan to discuss this idea with Infosight engineering team to incorporate this framework as a security solution for our customers.

References

1. NeDi - https://en.wikipedia.org/wiki/NeDi
2. Graph-based data clustering via multiscale community detection - https://appliednetsci.springeropen.com/articles/10.1007/s41109-019-0248-7 .
3. CASB – Cloud Access Security Broker - https://searchcloudsecurity.techtarget.com/definition/cloud-access-security-brokers-CABs
4. SNAP - Stanford Network Analysis Project - https://snap.stanford.edu/index.html