Darktrace Attack Path Modeling: Utilizing Graph Theory to derive multi-domain, risk-prioritized attack paths within computer networks*

February 23, 2022

Abstract

It is well known that the cyber security industry is both talent and resource starved. It is therefore critical that cyber security blue teams prioritize the defense of their networks to ensure maximum protection per unit cost. While red teams can provide insight into where effort and resource should be most immediately applied, the exercises themselves are often costly, non-exhaustive and infrequently run.

In this paper, the Attack Path modeling (APM) module of the Darktrace Prevent Framework is outlined. This technology sits at the core of the Darktrace Prevent product family, offering a real-time, automated, dual-aspect, multi-data-source, end-to-end attack-path-modeling capability. This module in particular, is designed to give blue teams a comprehensive view of realistic, risk-prioritized attack paths so that resources can be best allocated to defend key assets. As a proactive risk-reducing approach, this technology builds on Darktrace self-learning AI, an "engine" that produces continuously updated data for all assets across the entire digital domain.

In this paper, the internal aspect of the network takes the primary focus. Greater detail on other areas of the Darktrace Prevent product family will be discussed in separate, forthcoming literature.

1 Introduction

The use of Graph Theory[1] for APM could be used with some existing cyber security tools but these would be typically limited in their access to a specific type of data. For instance, despite proficiency in their respective fields, many will utilise a single data source such as Active Directory, while others may focus only on internal or external aspects of an organization.

However, a skilled adversary will aim to exploit vulnerabilities which span a variety of domains, covering both internal and external aspects of a target organization in order to achieve their goals. Access and seamless integration with this variety of data sources is necessary to create a realistic, end-to-end model of these attack paths. These include (but are not limited to):

- **Email** - In 2019, over 90% of attempted Cyber Threats were delivered by phishing email. [1]

*The author (who has chosen to remain anonymous) is based at the Darktrace AI Research Centre, Cambridge, UK and has 7 years of experience in the automation of complex cyber-centric processes with specialization in the offensive domain. The author also holds a PhD in Astrophysics from the University of Cambridge.

[1]Before going into more detail, some very basic terminology associated with Graph Theory is assumed – for further information refer to https://www.tutorialspoint.com/graph_theory/graph_theory_fundamentals.htm

- **Active Directory** – Essential for lateral movement within Windows environments.
- **SaaS / Cloud** - In 2021, 36% say their organization suffered a serious cloud security leak or a breach in the past year. [2]
- **Endpoint** – provides highly detailed data required for determining the defensive status of an endpoint.
- **Network** – very few attackers "land" exactly at their final objective. In general, they need to move laterally. Detailed knowledge of the network services, segmentation and normal network activity is critical to APM.
- **Vulnerability Management** - Up to one in three data breaches stemmed from unpatched software vulnerabilities in 2020 [3]. Comprehensive and up-to-date knowledge of both external and internal devices with unpatched vulnerabilities is critical as these are likely to be the first port-of-call for an attacker to probe.

Ignoring one or more of these aspects, will result in an incomplete evaluation of an organization's vulnerability to compromise and ultimately sub-optimal allocation of defensive resources and/or remediation efforts.

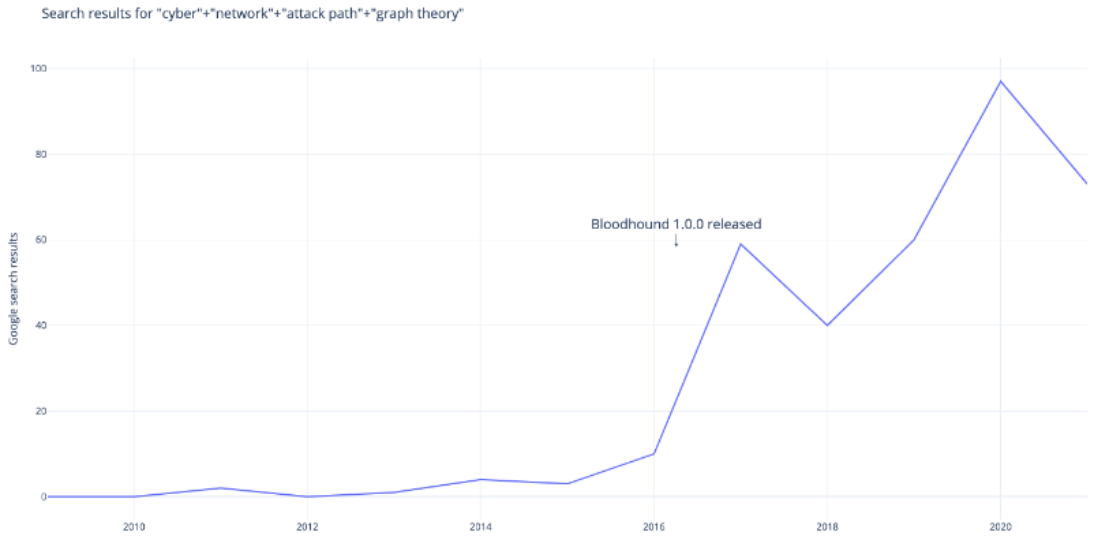

Figure 1: Number of Google search results as a function of time for the query: "cyber"+"network"+"attack path"+"graph theory"

2 Methodology

Darktrace Attack Path modeling prioritizes according to risk by assessing cyber-attack pathways, taking on the mindset of the adversary, probing the paths of least resistance. "Risk" in this approach can be defined as the product of two factors: event probability and event impact. The "risk matrix" (Fig. 2) is commonly represented visually as four quadrants with probability and impact assessed as low or high risk and ranked as minimal, intermediate or critical. The methodology draws on a wide variety of data sources, distilled by the machine learning engine, thus addressing potential vulnerabilities across all domains.

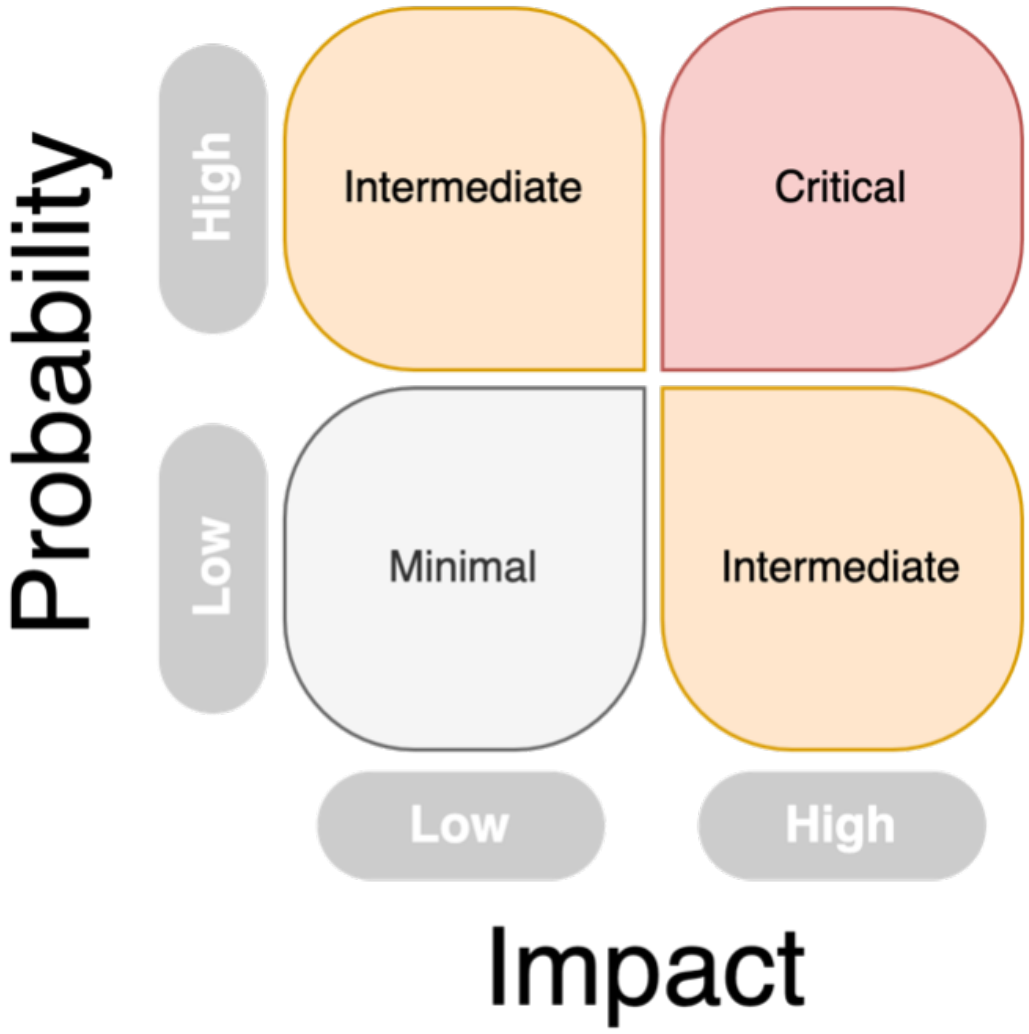

Figure 2: Visual representation of the Risk Matrix.

The fundamentals of our approach to risk-prioritized attack path modeling centre upon this principle and are represented by three main components, summarised in the three proceeding subsections.

2.1 Lateral Movement Probability Graph

Conceptually simple, this directed, weighted graph aims to estimate the **probability** that an adversary will be able to conduct successful lateral movement from node A to node B. Nodes are modelled as either devices or user accounts with a variety of attributes that influence the edge weight calculation. These weights are calculated based upon passively collected data where possible and actively collected where confidence of edge probability is low.

This evaluation of probability (represented by edge weight) takes into account a multitude of factors from a variety of data sources ranging from social engineering susceptibility to inferred likelihood of zero-day vulnerability development. Some examples of these and their corresponding

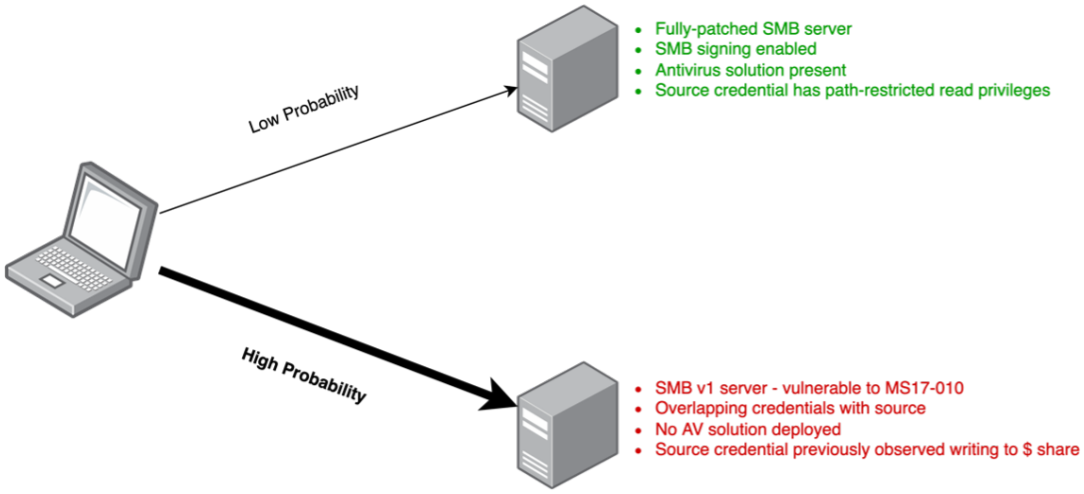

Figure 3: Basic principles associated with lateral movement probability estimation.

data source requirements are given below.

Internal Socially Engineered Spear-Phish (Data source: Email)

- Pre-existing regular communication between source and target.
- Precedent for sending potentially executable file types i.e. scripts, macro-enabled docs.
- Source holds senior position within organization to target.

Result: High likelihood of successful internal spear phish leading to code execution on target.

Poisoned SaaS Resource (Data source: SaaS)

- Source device SaaS credential has write privileges to shared SaaS directory.
- Shared SaaS directory contains executable file types.
- Some of these executable type files are regularly read by other SaaS users.

Result: High likelihood modified version of file will be executed by other SaaS users, resulting in code execution on their respective devices.

Broadcast Poisoning - capture authentication token (Data source: Network)

- Source device in same subnet as target.
- Target device observed utilizing broadcast name resolution protocol LLMNR.

- LLMNR hostname associated with SMB server.
- SMB Server code-signing is deactivated.

Result: High likelihood of intercepted LLMNR request from target leading to redirected SMB traffic to source and capture of authentication token.

2.2 Node Impact Score

The second component required for the risk calculation relates to **impact**. In contrast to the lateral movement probability, which is an **edge** property, impact is an intrinsic **node** property. Conceptually, this impact score should be representative of the resulting negative impact to the parent organization in the event that the node is compromised.

There are a variety of ways in which this impact score may be derived ranging from machine learning classification of sensitive SaaS and SMB resource paths to hierarchy analysis of user job roles from LDAP. Some examples of these methodologies are outlined below.

User / resource impact propagation

Imagine we have access to user job title or even hierarchy information via LDAP or similar. Our users can be "seeded" with an impact score – but what about the resources they access? What if we only trust the impact scores of a few users?

Figs. 4 and 5 illustrate a relatively simple approach to automated impact assignment, using propagation via shared resource access.

We begin with one high impact user – the CEO, and no prior information regarding the other users or files (assume no classification has been run on the filenames).

The fact that only one user (other than the CEO) has access to "sensitive.xlsx" implies that this file may be high impact. Furthermore, some of that importance is also propagated from the CEO to the one other user that also has access to "sensitive.xlsx".

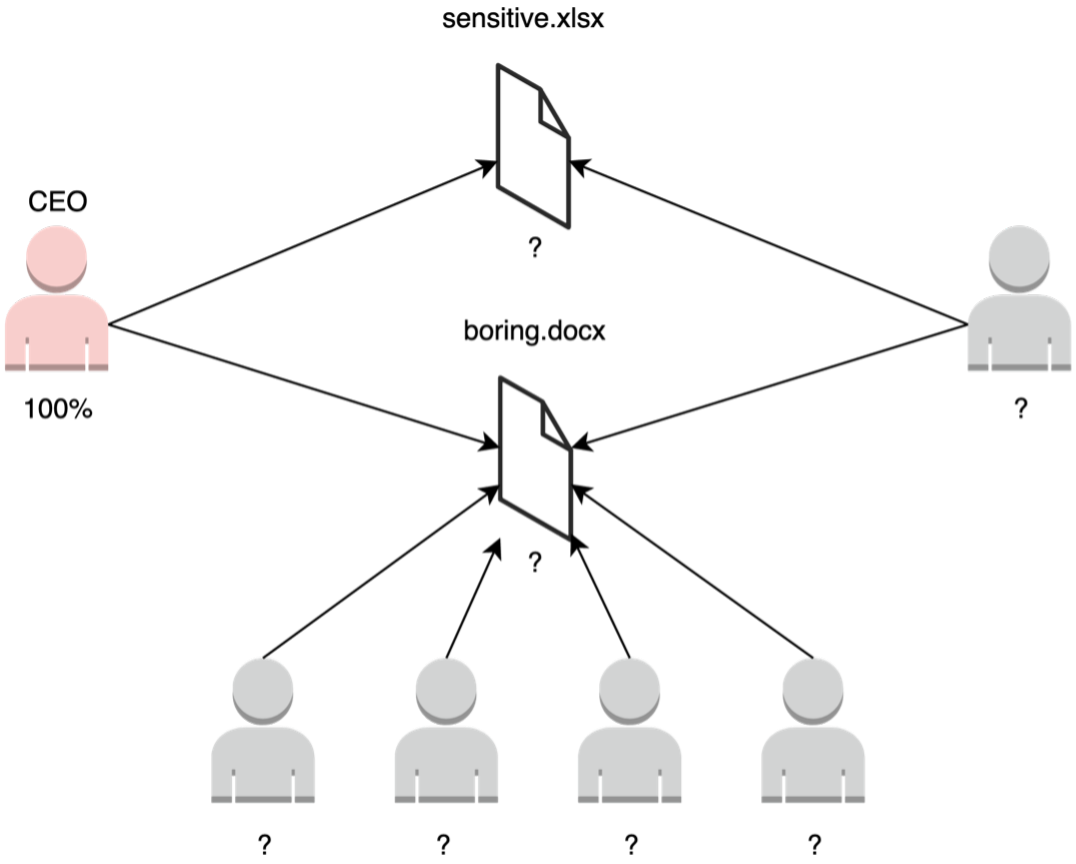

Figure 4: Visual representation of impact score pre-propagation.

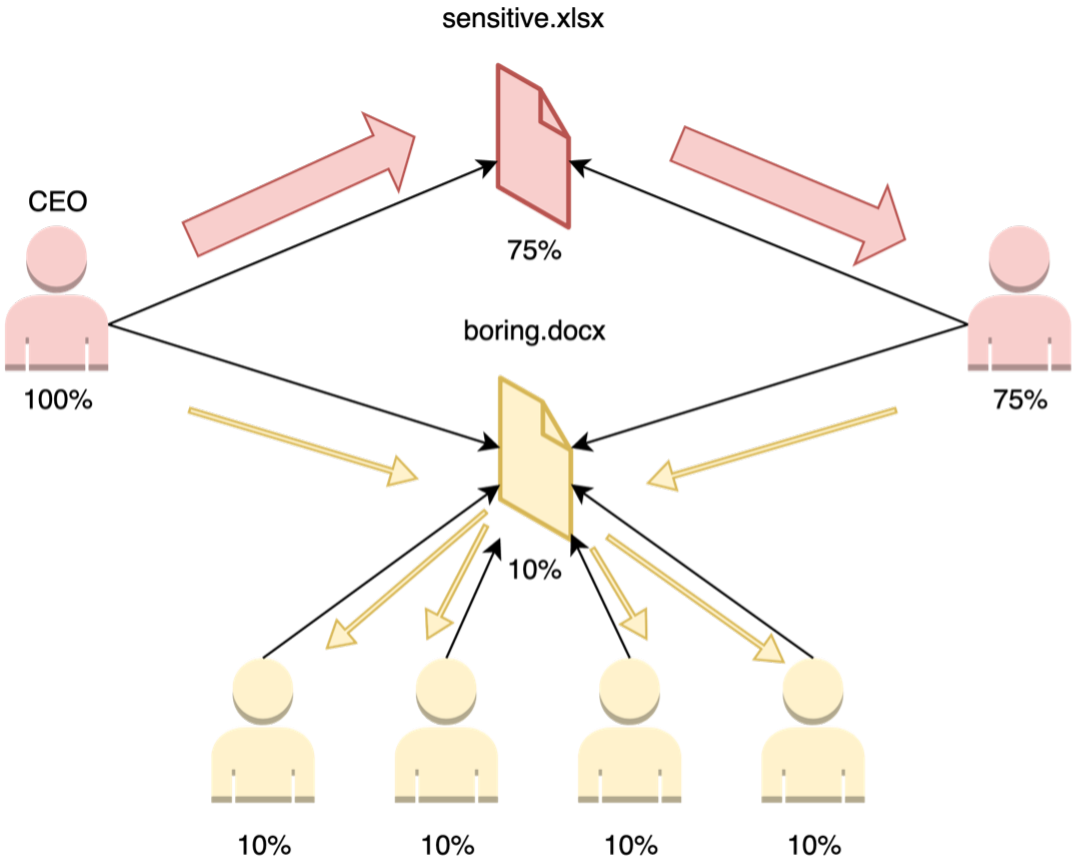

Figure 5: Visual representation of impact score post-propagation.

In contrast to this, "boring.docx" is accessed by a large number of users alongside the CEO. As a result, the impact propagation from the CEO is diluted by the large number of other unknown impact users also having access.

This impact propagation mechanism relies on the assumption of resource access segmentation assignment according to resource and/or user impact.

Key Server Identification

At an elemental level, one might state that:

> *An asset can be considered critical to a process if something is required from that asset in order for the process to continue as normal.*

This statement naturally extends into the domain of network assets – more specifically servers. If a significant number of client devices within an organization are retrieving (i.e. a data ratio in favor of download) data from a server, that server is likely to be critical to the organization. In other words, if that server was removed, the organization would not be able to function as normal.

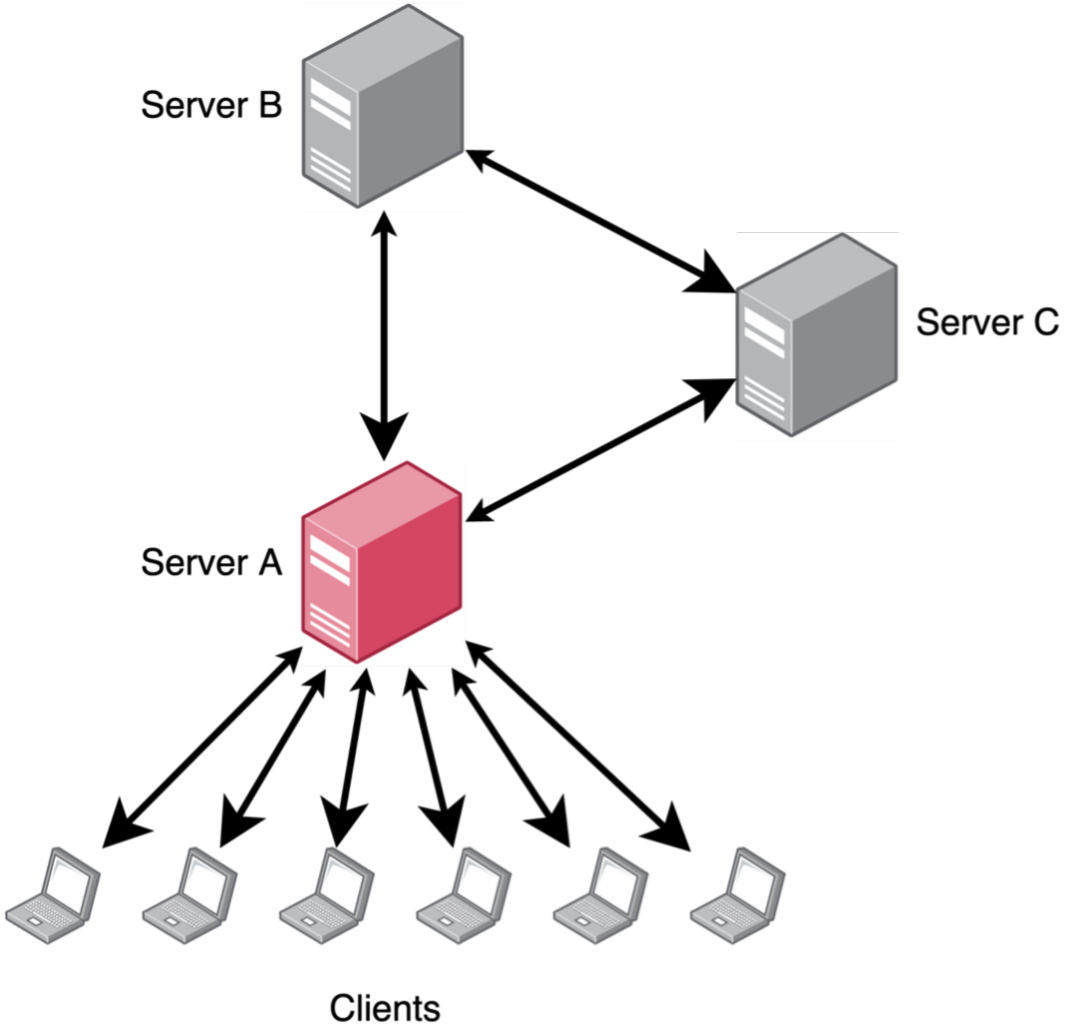

Figure 6: Key server identification from network traffic patterns. Server A is identified as a key server based on the large number of clients which depend on data from it. Note: asymmetrically sized arrows indicate dominant direction of data flow – not connection establishment.

However, in many cases, there exist additional servers, upon which these "tier 1" critical servers depend. This is visually represented in Fig. 6 – while Server A has been identified as a key server due to its high count of unique clients with download heavy data ratios, Clearly, Server B also provides data to Server A which is likely utilised in the process of delivering data to the clients. Consequently, impact from Server A is propagated to Server B as it appears likely that if Server B were removed, Server A would not function as normal and have an inability to provide data to the large client base.

In contrast, Server C has only 2 "clients" in this instance – Server A and Server B. Both are preferentially uploading data to Server C – which in this case may be a logging server or similar. At the data transfer level, Servers A and B do not appear to depend upon Server C's data for normal functionality.

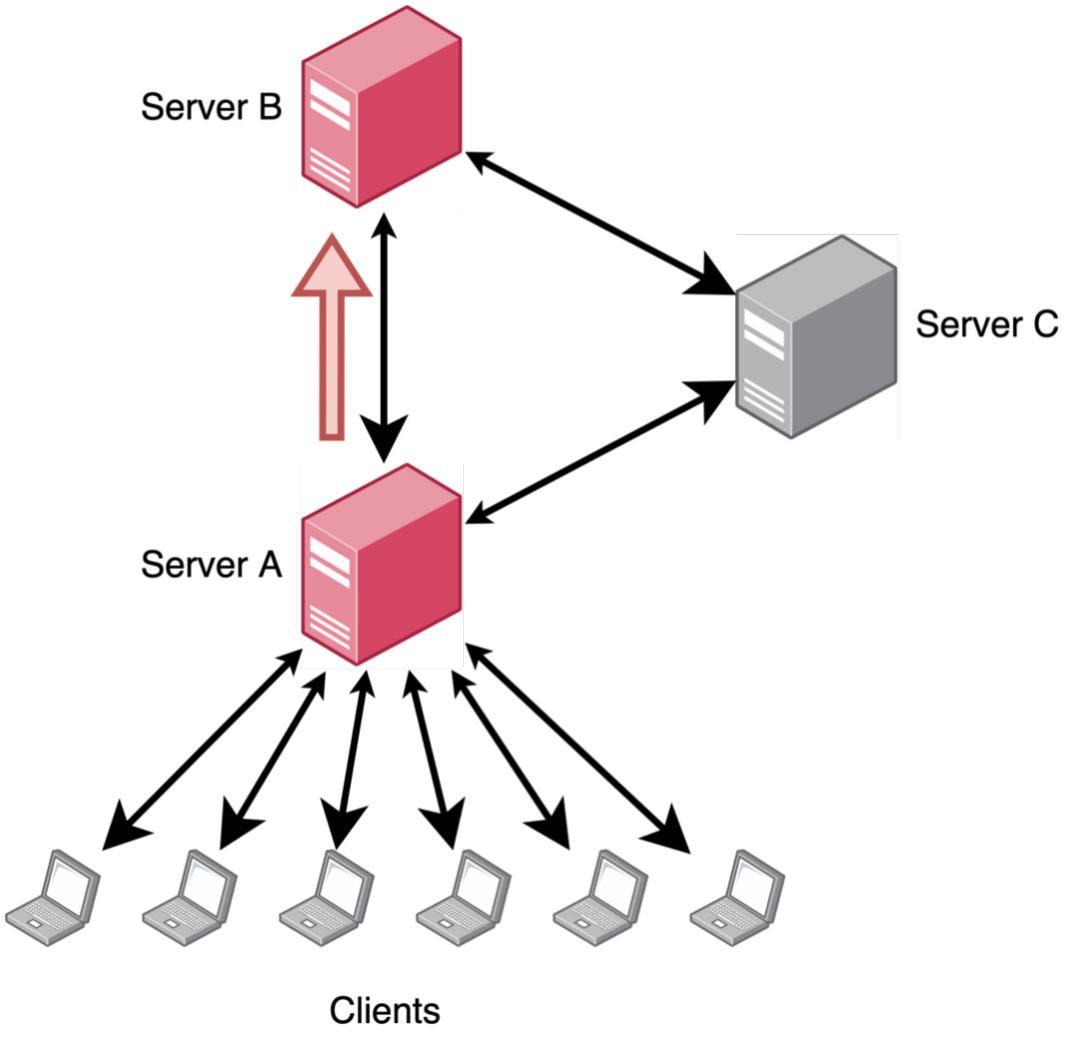

Figure 7: Key server identification and impact propagation via network patterns. Server A propagates impact score to Server B due to its perceived reliance on data from Server B. Impact is not propagated to Server C due to data ratios implying lack of dependency.

2.3 Ingress Probability

All attack paths require a start point. There are 5 main mechanisms for initial network penetration:

- **Phishing** – generally email based, but naturally extends to any instant messaging capability that is addressable by an unknown entity – Teams, LinkedIn, WhatsApp, SMS etc.
- **Exploit externally facing server** – once a new vulnerability is identified, attackers will utilise full IPv4 scan data to identify valid targets for exploitation.
- **Drive-by exploit** – typically associated with malicious re-direction from already questionable websites. Reached heights of popularity with the "BlackHole" exploit kit – the creators of which were caught in 2016 [4].

- **Insider** – for organizations with very strong perimeter defenses, this may well be the most viable method of ingress for a determined threat actor.
- **Third-party-compromise** – comparable to the Insider ingress but in device/software form as opposed to a human.

In principle, any human controlling a device with internet access is vulnerable to social engineering. Additionally, any device which is externally (Internet) facing, should also be considered a potential ingress point.

However, there are a variety of factors which might influence the probability of ingress for a user or externally facing server, some examples are given below:

User

- **Results of previous red-team assessment** – If a user was previously successfully phished, one may assume that they retain that proven susceptibility. Research indicates that awareness may temporarily spike post-assessment, but eventually returns back to pre-assessment levels.
- **User email exposure / perceived target value** – This can be estimated by monitoring the number of phishing emails targeting the user. Typically, these will be an intersection between perceived high privileges and low technical knowledge, such as a CEO's personal secretary.
- **Patch-level of the user devices** – If a user does not regularly run software updates, their security awareness level is likely to be lower than someone who does.
- **User web-browsing habits** – if the user frequents low-trust endpoints, they are more liable to malvertising redirection.

Externally-facing-server

- **Patch-level** – if an external server remains unpatched, they are highly likely to be exploited.
- **Service Port** – if the service is running on a non-standard port, the presence of the server is less likely to be detected during full IPv4 scans.

The inferred ingress probability associated with the starting node allows us to modulate the overall probability associated with a given attack path, yielding a more realistic, risk-prioritized output.

3 Execution

With representative estimates of the lateral movement probability graph, node impact scores and ingress probabilities, one can finally execute the attack path modeling simulation.

The simulation is run in the following manner:

- Threshold is applied to node importance values in order to determine target nodes.
- Dijkstra's algorithm [5] is utilised to calculate shortest paths from all possible ingress nodes to all target nodes.
- The attack paths are then weighted according to total impact per unit path length and modulated according to ingress probability associated with the start node.

4 Remediation

The modeling process produces a comprehensive set of risk-prioritized attack paths, giving the cyber security team the opportunity to evaluate how best to use this new information. Further simulation can be run to identify the edges which, if neutralised, would minimise the total risk of all derived attack paths. The extensive visibility and breadth of data sources leveraged by the Darktrace APM module to identify these attack paths can also be used to instantly apply enhanced Detect and Respond capabilities to defend them. Such measures can greatly assist the blue team by providing behaviorally-defined, precision-molded armor while more comprehensive patching takes place against these high-risk nodes and edges.

5 Conclusion

In a resource starved cyber security industry, blue teams require realistic and continuous evaluation of their digital estate. The necessity of data source breadth is discussed and the underlying principles of risk along with methods to achieve accurate quantitative estimation via Graph Theory methods are demonstrated. Made possible only by the breadth of data sources, subsequent distillation through the power of machine learning and fusion with Graph Theory, Darktrace APM signifies a point of inflection, a return of hope to the beleaguered blue components of the cyber security world.

References

[1] TrendMicro. Online phishing: How to stay out of the hackers' nets. https://news.trendmicro.com/2019/11/20/online-phishing-how-to-stay-out-of-the-hackers-nets/, 2019.

[2] Fugue. The state of cloud security 2021 report. https://resources.fugue.co/state-of-cloud-security-2021-report, 2021.

[3] IBM-Security. X-force threat intelligence index. https://www.ibm.com/downloads/cas/M1X3B7QG, 2021.

[4] Brian Krebs. 'blackhole' exploit kit author gets 7 years. https://krebsonsecurity.com/2016/04/blackhole-exploit-kit-author-gets-8-years/, April 2016.

[5] Dijkstra, Edsger W. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.