The Dream of a Red King
A Connection Protocol for Small-World P2P Botnets

Jacob Moore

19 November 2023

Disclaimer

By continuing to read this article, you acknowledge that you accept the responsibility to use the information contained herein in an ethical and legal manner. The author of this article strongly advocates for the ethical use of cybersecurity knowledge and disclaims any support for illegal activities. This article is intended for educational and ethical purposes, aiming to enhance the knowledge of cybersecurity professionals. Readers are encouraged to use this information responsibly and within legal and ethical boundaries. The author holds no liability for any actions taken based on this information.

Introduction

This research was motivated by the final chapter in VX-Underground's *Black Mass: Volume I*, authored by b0t and titled "The RedKing Hivemind." The chapter delves into the concept of establishing a decentralized P2P botnet through small-world networks. Intrigued by this exploration, I took on the ambitious challenge of implementing the infrastructure code for a "Red King" network myself. In all honesty, I woefully underestimated the complexity and magnitude required for such an undertaking. While I may not have attained the creation of the Red King, I believe I have successfully devised a connection protocol tailored for small-world networks that aligns with the outlined objectives in "The RedKing Hivemind." The research found herein is some of the fruit from that labor.

While the foundation of this research is from the perspective of a malware developer, it's important to recognize its broader applicability to any system utilizing small-world networks. This extends beyond the realm of malware, encompassing diverse models ranging from power grids to protein-to-protein interaction networks. Approach this content with an open mind and a creative spirit to fully grasp its interdisciplinary relevance.

Before proceeding, I strongly recommend reading the chapter to gain a better understanding of these types of networks and how they could be leveraged by botnets. The entire book was authored by a group of remarkable and passionate individuals and is a delightful reading experience if you enjoy cybersecurity. VX-Underground offers the book for free on their website or a physical copy can be purchase on Amazon.
Black Mass PDF: VX-Underground—Black Mass
Black Mass Paperback: Amazon —Black Mass

Code and 2D/3D network models can be found in the research's Github repository at https://github.com/moorejacob2017/RedKing_ConnProto/

Please enjoy!

Objectives

The objectives of this research are straightforward. P2P networks often face challenges with an excessive dependence on command and control servers and super nodes, rendering them vulnerable to node enumeration attacks that extract the locations of all bots. Integrating a small-world network design in the system aims to maintain decentralization without relying on servers or super nodes to relay instructions to the bots. Small-world networks seek to limit the number of connections a node can establish and restricts the scope of these connections to local nodes. Occasionally, nodes are permitted to establish long-distance connections across the network, allowing for fewer hops during routing.

These networks also offer significant redundancy, enhancing resilience against network failures. While these principles hold promise in theory, the challenge of the implementation lies in establishing node connections that effectively thwart node enumeration attacks.

Given the decentralized nature of the network, nodes must assume the responsibility of networking, requiring access to specific information about other nodes, routing, and the overall network. To prevent enumeration, the design must limit the amount of information nodes have about the network while still being able to carry out network operation effectively. To streamline the discussion, this paper will specifically address the connection protocol for integrating nodes into the network. The objectives are outlined as follows:

1. Create a connection protocol for a small-world P2P botnet.
2. Design both the network and connection protocol to resist node enumeration attacks.
3. Design the network with as much redundancy as possible to prevent outages during botnet take-downs and node failures.

Characteristics of Small-World Networks

Pause for a moment to reflect on an person's social circles. It's common for a person's friends to share mutual connections within the same friend group. Yet, consider how a person may also have distinct friend groups associated with work, school, and hobbies. This type of interconnectedness helps to illustrate the "small-world phenomenon," and is an example of a small-world network, wherein there are clusters of mutual connections linked by occasional longer-distance relationships.

Small-world networks exhibit a blend of local clustering and global connectivity. They possess a balance between order and randomness, yet have an overall structure that allows for efficient short-path communication across the entire network. These networks can be identified by having the following characteristics:

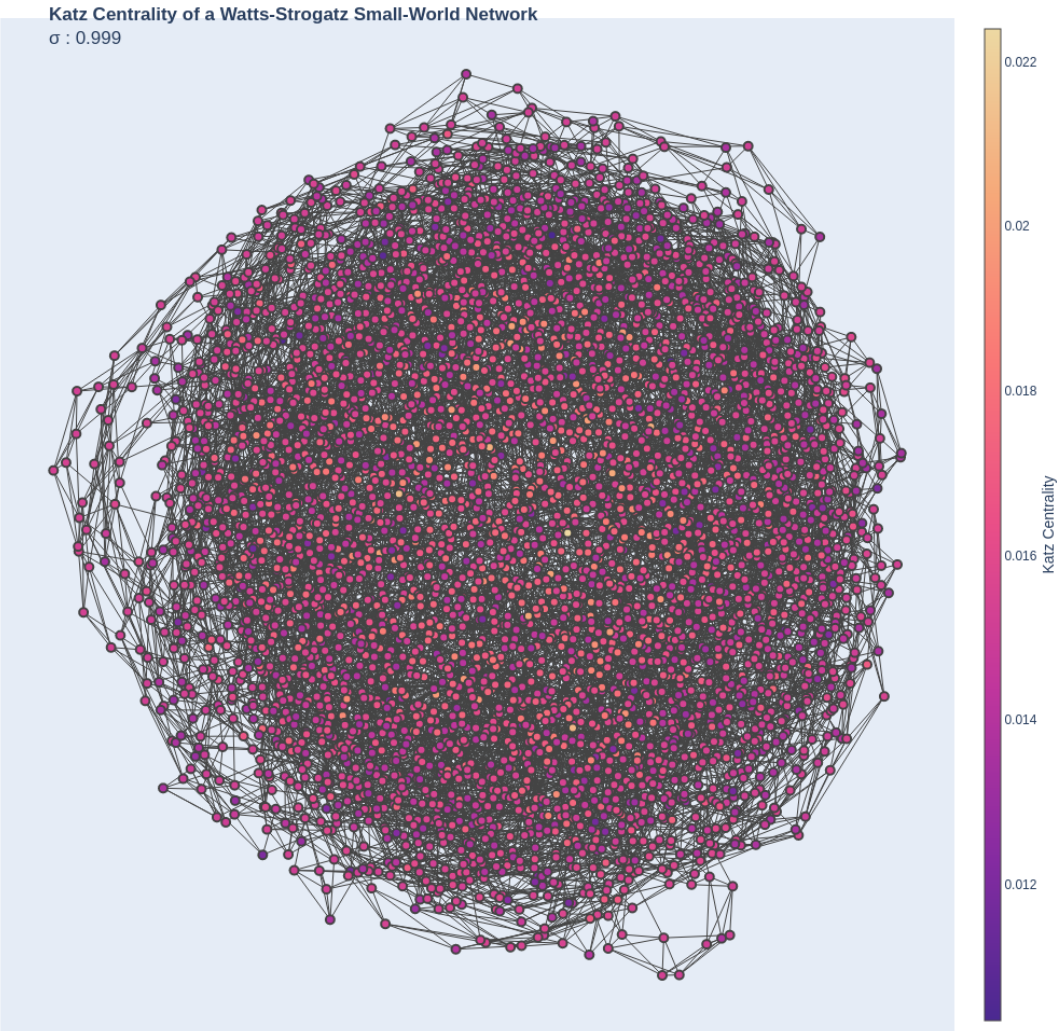
- Small-world networks tend to have a large cluster coefficient. The cluster coefficient measures how often nodes share the same neighbors, creating clusters in the network. In the context of small-world networks, attaining a cluster coefficient of 0.5 serves as a sound and appropriate target.
- Small-world networks also have relatively small path lengths between any 2 nodes in the network. The average shortest path of any given small-world network grows proportional to the logarithm of the number of nodes in the network ($L \propto \log N$). That means that as the number of nodes increases, the average shortest path will increase logarithmically and will produce a logarithmic graph.
- The small-worldness of a graph can be measured as its small-coefficient (σ). A small-coefficient compares a graphs cluster coefficient and average shortest path length against a random graph. In this paper, the random graph will be a Watts-Strogatz small-world network and a proper small-world network will result in a small-coefficient greater than or approximately equal to 1. While the small-coefficient is not always the best way to measure small-worldness, it will suffice for the purposes of this research.

It is important to note that for all of the network models illustrated in this paper, node coloring is determined using Katz centrality. Katz centrality serves as a metric for assessing a node's significance within the network. Nodes with higher centrality levels are represented with brighter colors, indicating their increased importance to the network.

The Watts-Strogatz Small-World Network

Discovered by Duncan Watts and Steven Strogatz, a Watts-Strogatz network is initially generated by establishing a ring of nodes and connecting each node with its neighbors. Once finished, all connections have a probability of being replaced with a link to a randomly chosen node. While these networks exhibit small-world characteristics, their generation requires a pre-existing network. A network is first created and then modified into a small-world configuration by systematically visiting each node and considering the possibility of rewiring it to a random node. This process necessitates the enumeration of every node, a task we aim to avoid explicitly. Moreover, it depends on awareness regarding the presence of nodes beyond neighboring connections during rewiring instances, a vulnerability that could be exploited in enumeration attacks. These aspects of the generation process prove impractical for the gradual addition of nodes to a small-world P2P botnet and are in direct opposition to the goal of enumeration resistance. This helps to answer the question as to why the conventional generation of small-world networks proves inadequate for achieving the goals outlined earlier.

While a Watts-Strogatz network may not be suitable for constructing a small-world botnet, its desirable traits make it well-suited for comparing the small-worldness of other networks. Therefore, this type of network will be used to calculate the small-coefficient of the networks created using the protocol detailed in this research. An example of a Watts-Strogatz small-world network is provided below.



Red King Network Assumptions

For a variety of reasons, a few assumptions about the network and its nodes will be made to provide a necessary simplification of complex real-world scenarios. These assumptions foster consistency and comparability between generated networks, and allow for meaningful comparisons and analyses across different scenarios. While these assumptions may not capture every nuance of reality, they serve as pragmatic adaptations for modeling, striking a balance between accuracy and practical feasibility.

1. **Assumption of Self-Replication**: Nodes are produced by other nodes in an asexual fashion, similar to that of a computer virus or worm.
2. **Assumption of Reproduction Rate**: All nodes reproduce on cycles at a fixed rate of one new node per existing node per cycle.
3. **Assumption of Non-Dropping**: Nodes are not dropped from the network.

One of the products of these assumptions is that the sum network growth will be exponential, with 2^x nodes at cycle *x*. For example, Cycle 3 will have 2^3 (8) nodes, Cycle 4 will have 2^4 (16) nodes, Cycle 5 will have 2^5 (32) nodes and so on. This allows the generated networks to grow in a uniform fashion and makes analysis easier.

The Red King Connection Protocol

The presented connection protocol is intentionally designed to resist node enumeration and is tailored specifically for use in small-world networks. Its purpose is to systematically incorporate nodes into a small-world network, ensuring that each node maintains a maximum number of neighbors while minimizing the disclosure of additional nodes as much as possible. Keep in mind that the ensuing research focuses solely on the connection protocol, excluding considerations for the overlaying node communication protocols.

The protocol can be broken down into two primary components. The first part focuses on establishing node connections within the network, while the subsequent part addresses the integration of new nodes and the idea of node discovery.

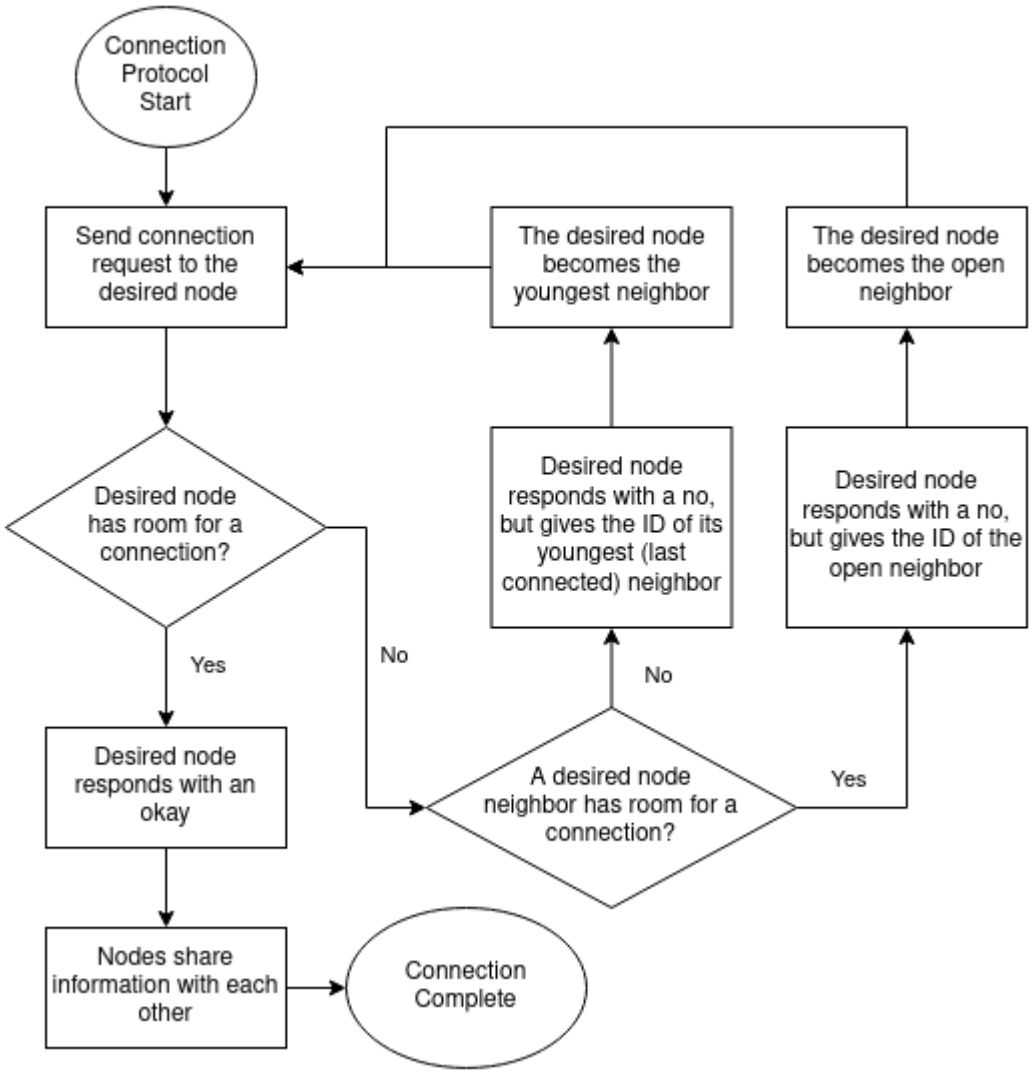
Part 1: Node Connection
The node connection protocol is as follows:

1. Node A wishes to connect to Node B, which is already established in the network. Node A sends a connection request to Node B.
2. Node B will give a response based on one of three conditions…
   – **Condition 1**: If Node B has not reached is maximum number of connections, it will respond with a confirmation and agree to connect with Node A.
   – **Condition 2**: If Node B has reached its maximum number of connections, but is aware of one of its neighbors that has not reached its maximum number of connections, it will respond with a rejection to connect, but give Node A the location of the neighbor that has not reached its maximum number of connections.
   – **Condition 3**: If Node B has reached its maximum number of connections and all of Node B's neighbors have also reached their maximum number of connections, then Node B will respond with a rejection to connect, but give Node A the location of the last neighbor it connected with, AKA. its "youngest" neighbor.
3. Upon receiving a response from Node B, Node A will attempt to repeat the connection process with any node location given in the response of Node B until it has found a node that has agreed

to connect. In the event that Node B agrees to connect, Node A will continue the connection process with Node B.

4. Once Node A has found a node that has agreed to connect, both nodes exchange any networking information needed and consider each other as neighbors.
5. The connection process is complete and Node A is considered to be connected to the network.

The given connection protocol has been illustrated in the following flow chart for readability and for ease of understanding.



Part 2: Node Discovery

In an optimal scenario, newly created nodes are equipped with information about the locations of three other nodes. Aligning with the assumption of self-replication, these three nodes consist of the parent node, the parent node's second youngest neighbor, and the parent node's third youngest neighbor. The exclusion of the youngest neighbor is deliberate, given its role in network connection. Embedding the

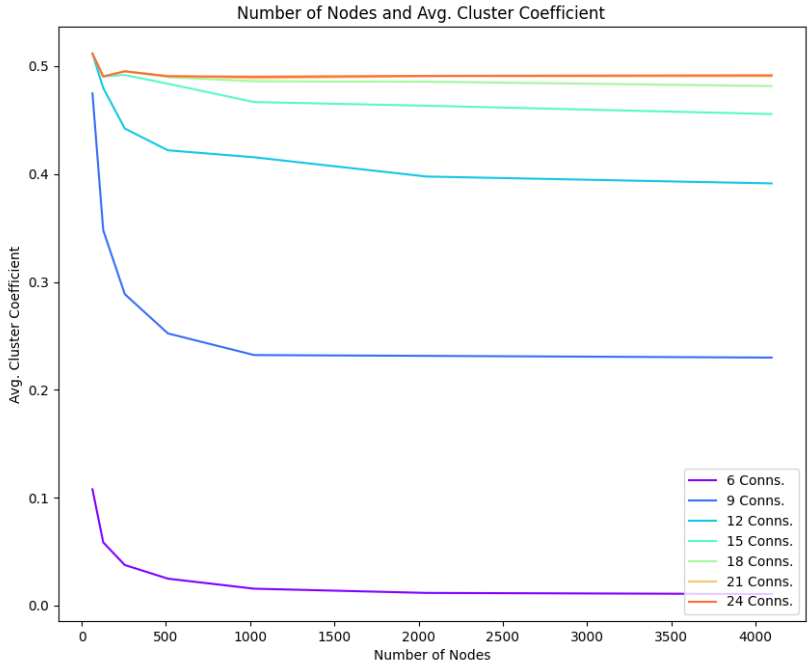
youngest neighbor as a default connection could potentially lead to self-connections and create vulnerabilities in the network.

While having embedded locations is ideal, it may not be entirely practical in all situations. Hence, it might be appropriate to allow for the disclosure of the second and third youngest neighbors upon request. Although this might appear counter-intuitive to the goal of enumeration resistance, its acceptance will be deliberated further when discussing the rationale behind specific design choices.

Proof of Small-Worldness

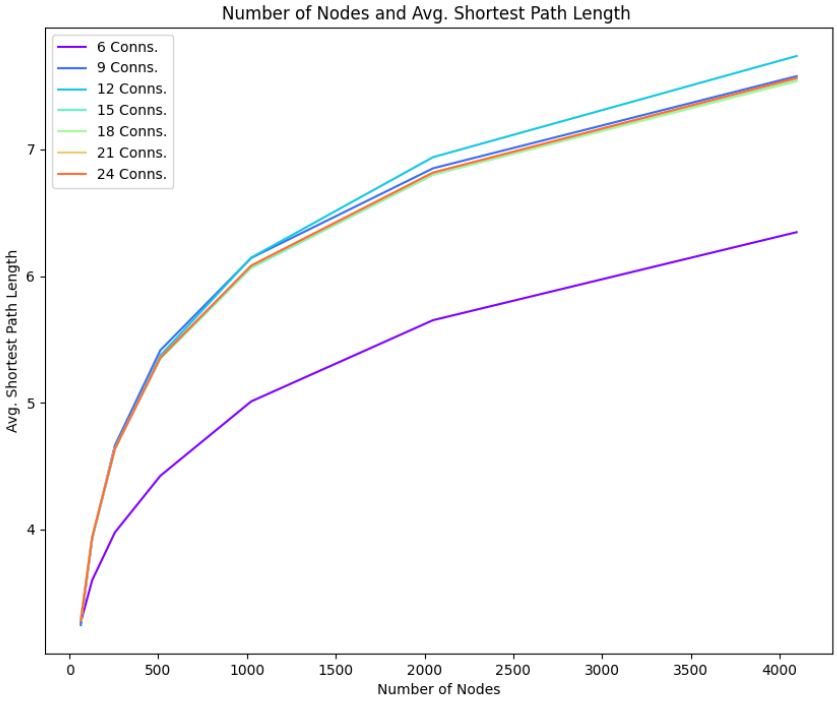
Before discussing the design choices of the protocol, let's first establish that the protocol indeed generates a small-world network. As previously mentioned, three criteria can be employed to determine if a network possesses small-world properties: the cluster coefficient, the average shortest path length in relation to the number of nodes, and the small-coefficient.

Displayed below is a graph illustrating the average cluster coefficient of networks generated by the protocol in relation to both the network's size and the maximum number of connections per node (abbreviated as "Conns."). The graph shows that as the number of the network's nodes increases, the cluster coefficient stabilizes. Additionally, it demonstrates that with an increasing number of connections, the cluster coefficient approaches the targeted value of 0.5. However, the pace at which the cluster coefficient converges to 0.5 decreases substantially as the maximum number of connections increases. Therefore, achieving a balance between the cluster coefficient and the maximum number of connections is crucial. Following testing, it was determined that a cap of 15 connections per node proved to be optimal for the network.

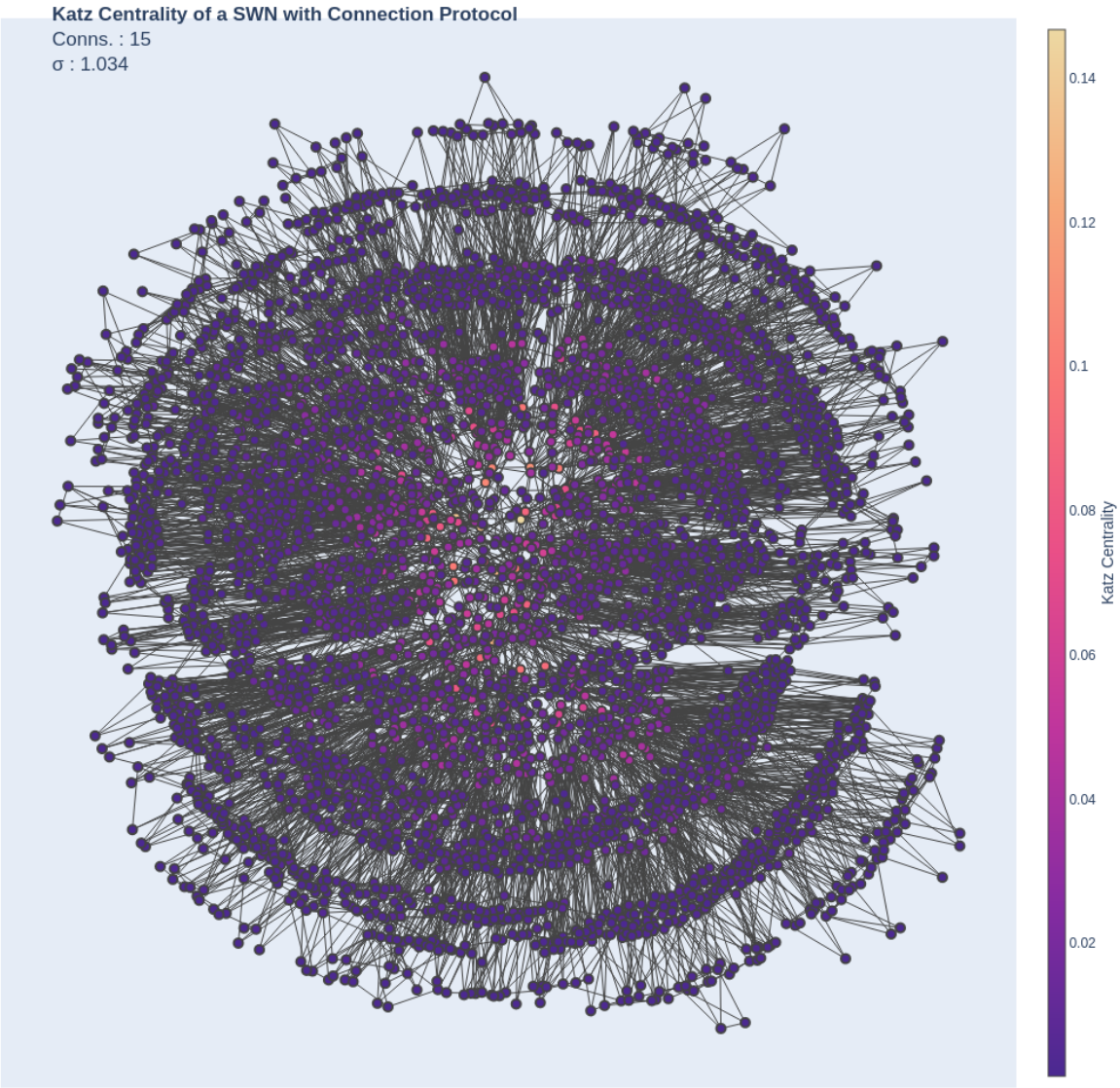


In general, the average shortest path length remained relatively stable with minimal fluctuation across varying maximum node connection capacities. Although the graph may appear uneventful, it provides

crucial evidence supporting the creation of small-world networks. The growth curve of the average shortest path length in relation to the number of nodes follows a classic logarithmic pattern, exemplifying the proportional property found in small-world networks. It is important to highlight that the line representing the 6-connection networks is *not* a result from a relation between the decrease in average path length and the decrease of the maximum number of connections. Instead, it constitutes an anomaly generated by the protocol, which will be further discussed in a subsequent section.



Finally, a network was generated using the connection protocol, and the small-coefficient was calculated by comparing it to a random Watts-Strogatz network of a similar size, akin to the previously presented example. The Red King network was created with a node connection capacity of 15 connections and ran through 12 cycles, resulting in 4096 nodes. The obtained small-coefficient for the network was 1.034, categorizing it as a small-world network.



Protocol Design and Properties

The Red King connection protocol generates small-world networks in a fashion such that the network state is independent of node additions, instead relying on node states. These networks can be characterized by a greater dependence on older nodes within the network. This dependency is evident in the higher Kats centrality of the central nodes compared to the younger nodes situated on the periphery. While the network is technically decentralized, the removal of old nodes at the core of the network would disproportionately impact the network's performance and coherence.

In a preliminary resiliency test, the network underwent a stress test that systematically removed the oldest nodes until a network segment became disconnected from the main network. It's essential to acknowledge a limitation of this test, as a lost network segment can be as small as a single node and may not provide a comprehensive assessment of the entire network's resilience. Following the stress test, it was determined that the network could endure the removal of up to 6.25% of its oldest nodes,

equivalent to 256 nodes out of a total size of 4096. While not the best performance for its purpose, this result indicates the presence of redundancy within the network, demonstrating its capacity to withstand minor perturbations.

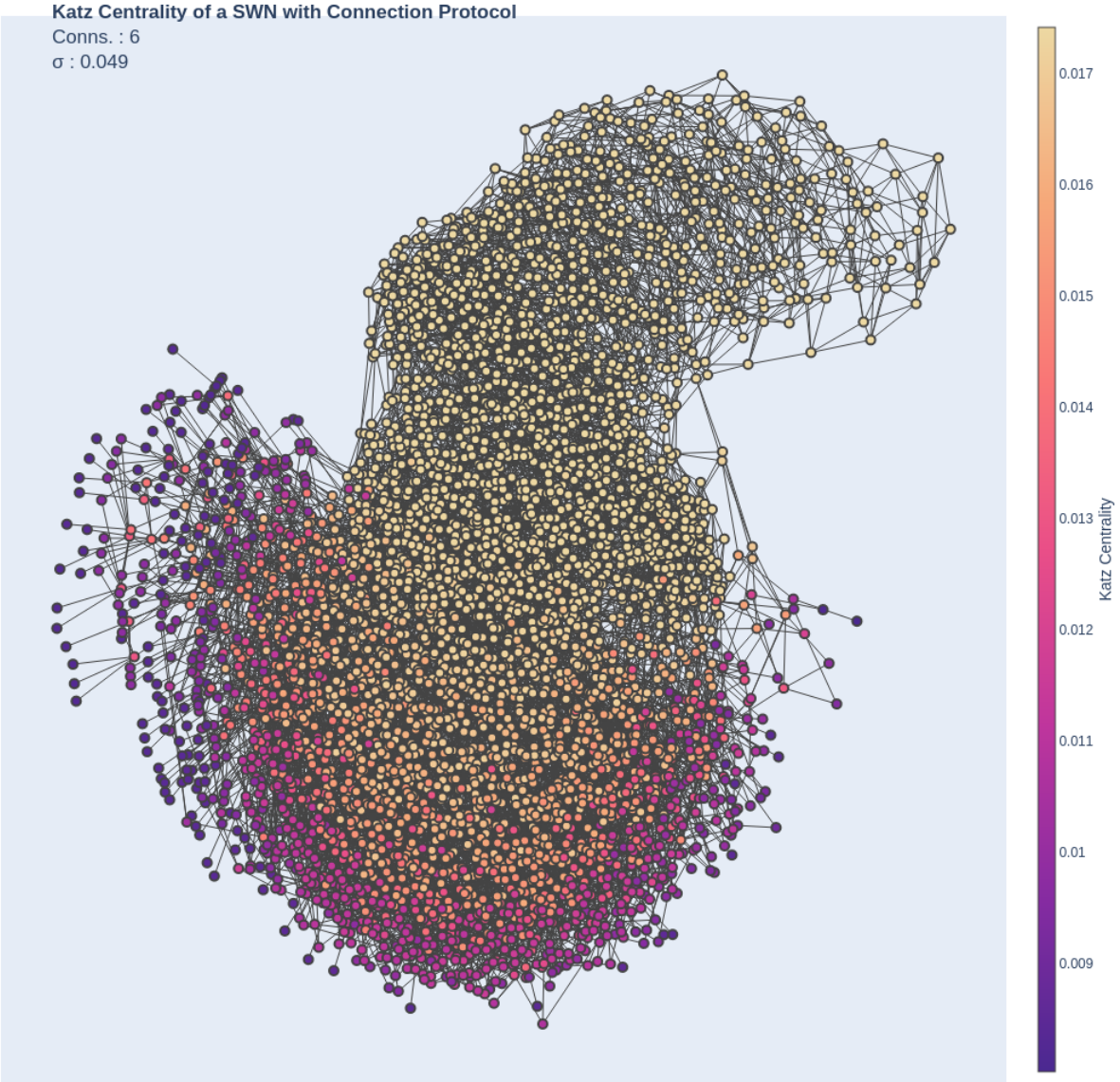
Given the dependence on older nodes, the protocol intentionally organizes and guides the network to obscure the oldest nodes, making them more challenging to access. Observe in the model how newer nodes form layers that encapsulate the network core. This structural arrangement is attributed to the protocol's deliberate provision of the three youngest neighbors for any given node. If the protocol were to provide neighbors randomly, nodes could be enumerated through repeated connection requests until all neighbors are revealed. By restricting the neighbors provided during network connection, the older nodes retain a level of obscurity.

As younger nodes are considered more expendable than their older counterparts, they can be disclosed during enumeration attacks, protecting the older nodes from detection and exposure during botnet take-downs. To coax a node to disclose its oldest connections, the most recent connections must be taken offline first, followed by re-enumerating the target node. This effect is compounded with each additional "layer" that is wrapped around the network core, significantly complicating enumeration attempts and thwarting attacks that exploit the connection protocol. It is important to note, however, that while the connection protocol provides resistance, the network is not entirely immune. Communication, if designed or implemented poorly, could still be leveraged for enumeration and attacks.

A number of modifications can be implemented in or appended to the connection protocol. For instance, the idea of introducing a random chance during node creation to increase the maximum number of connections for the new node was considered. There is ample space for refinement and additional experimentation to discover the characteristics that contribute the most to network resiliency.

The Anomaly of 6-Connection Networks

During testing, a peculiar anomaly emerged in the data sets, revealing intriguing characteristics. When employing the connection protocol with a node connection limit set at 6 connections, the network adopts an irregular structure with distinctive properties not observed in prior presentations. It's crucial to emphasize that this phenomenon is exclusive to the 6-connection cap, as all other tested connection capacities, including 5 and 7, exhibited the expected behavior. In the spirit of full transparency, the underlying cause of this phenomenon within the protocol at this specific capacity is not fully understood, necessitating further study for clarification.



As observed earlier, the network with a 6-connection capacity exhibits an exceptionally low cluster coefficient, but boasts a more efficient average shortest path compared to other networks. Consequently, it yields a subpar small-coefficient (0.049), falling short of the criteria for a small-world network. Nonetheless, it theoretically retains the same enumeration resistance endowed by the connection protocol, featuring a “front” of younger nodes that propagates unidirectionally rather than radiating outward in all directions. Furthermore, it maintains a lower average node centrality, enhancing decentralization and displaying remarkable resilience by accommodating the removal of up to 24.95% of the network’s oldest nodes without experiencing a network segment disconnection. This translates to the removal of 1022 nodes from a network of 4096 without any segment loss. Although the 6-connection network does not qualify as a small-world network, it demonstrates all the desired traits for a P2P botnet resistant to node enumeration, potentially surpassing the small-world networks examined in this paper. However, further research is required to substantiate this potential superiority.

Conclusion

In conclusion, the Red King Connection Protocol presents a novel approach to establishing and maintaining connections in a network, showcasing its effectiveness in generating small-world networks. The proof of small-worldness, illustrated through comprehensive testing, demonstrates the protocol's ability to create networks with desired properties. It is further validated by its comparisons to Watts-Strogatz networks, firmly establishing it as a small-world network.

Moreover, the protocol's unique features contribute to the resilience of the network, as demonstrated in stress tests and enumeration resistance mechanisms. The intentional organization of the network, with a dependence on older nodes and a strategic arrangement adds a layer of network obscurity. The stress tests revealed the network's capability to endure the removal of a portion of its oldest nodes, showcasing its redundancy.

Additionally, the anomaly observed in 6-connection networks presents intriguing characteristics, showcasing potential resilience in the face of node enumeration. However, further research is warranted to fully understand and substantiate the exceptional traits of the 6-connection network. In essence, the Red King Connection Protocol provides a foundation for building resilient networks, offering a valuable contribution to the field of network design and security.

Citation

- Aric A. Hagberg, Daniel A. Schult and Pieter J. Swart, “Exploring network structure, dynamics, and function using NetworkX”, in Proceedings of the 7th Python in Science Conference (SciPy2008), Gäel Varoquaux, Travis Vaught, and Jarrod Millman (Eds), (Pasadena, CA USA), pp. 11–15, Aug 2008
- b0t, “The RedKing Hivemind.” *Black Mass*, edited by smelly_vx, vol. 1, VX-Underground, Monee, IL, 2023, pp. 89–96.
- Mason A. Porter (2012), Small-world network. Scholarpedia, 7(2):1739., revision #198607
- “Small-World Network.” *Wikipedia*, Wikimedia Foundation, 5 Nov. 2023, en.wikipedia.org/wiki/Small-world_network.