The Academic
International Journal of Multidisciplinary Research

An Online Peer Reviewed / Refereed Journal
Volume 1 | Issue 5 | December 2023
ISSN: 2583-973X
Website: www.theacademic.in

Securing Spark Cluster by Detecting and Eliminating Malicious Node Using DNA Cryptography

Dr. J Balaraju, M.Tech., Ph.D., Assistant Professor, CSE, Anurag University

***K V Sathwik**, Student at Anurag University

R Nikhil, Student at Anurag University

K Akshaya, Student at Anurag University

ARTICLE DETAILS

Research Paper

Keywords :
DeoxyriboNucleic Acid, Media Access Control, Dynamic Host, Configuration Protocol

ABSTRACT

Apache Spark has a scalable, and efficient disbursed machine supporting commodity hardware with the aid of combining specific networks within the topographical locality. Node support inside the Spark cluster is unexpectedly growing in one-of-a-kind versions which can be dealing with issues to control clusters. Node identification in a cluster absolutely depends on DHCP servers that handle IP addresses, hostname is based totally at the physical address (MAC) address of every node. There is a scope for the hacker to theft the information using IP or hostname. We are proposing new node detecting mechanism for avoiding adding duplicate node into the cluster. In this we used DNA hiding method to producing a completely unique hostname with the combination of unique Physical address (MAC) of node, hostname and IP. This mechanism is supplying higher node control for the Spark cluster offering adding and deletion node mechanism through the use of constrained computations and providing better node security from hackers. The main objective of this project is to design a set of rules to put in force node touchy records hiding using DNA sequences and supplying safety to the node and its data from hackers.

1. INTRODUCTION:

Overview

Node management in a Spark cluster refers to the process of efficiently handling and coordinating the various nodes (machines) within the cluster to ensure smooth and effective execution of data processing

tasks. Efficient node management guarantees resource optimization, effective task distribution, and reliable data processing in a Spark cluster.

We proposing novel node management approach addresses the growing need for safeguarding sensitive data.

What is it?

Node management in a Spark cluster refers to the process of efficiently handling and coordinating the various nodes (machines) and to stop the services to duplicate hostnames thus ensuring security.

Why is it needed?

Apache Spark Cluster node management is a crucial issue, and any Spark versions not having its own security mechanisms, especially node management. So, we provide a layer on the top of cluster to stop services to malicious/duplicate nodes.

Applications:

Malicious Node Detection and elimination: Malicious node can be prevented entering into cluster because of dynamic change of unique hostname by DNA cryptography.

Dynamic Key Rotation: DNA-based keys can be rotated periodically, making it even more challenging for malicious nodes to gain unauthorized access or stay undetected within the cluster.

Problem Statement

Securing a Spark Cluster against malicious nodes and ensuring the integrity and authenticity of data within the cluster using DNA Cryptography. Node identification is very important to start spark daemons for running spark cluster and it is not possible to run with duplicate hostnames for secure data integrity.

2. PROPOSED METHOD:

We are change the hostname to unique key to start the daemons thereby this key generation will result to stop the services to a duplicate node.

How DNA Cryptography Works in proposed method:

You create a unique DNA-based identifier for each node in the cluster based on its hostname, ip-address and mac-address.

Initial Setup:

Imagine a Spark Cluster with multiple nodes, each identified by a hostname.

Each node generates a unique DNA-based key from its hostname, ipaddress and mac-address using DNA Cryptography.

Unique Features:

The DNA identifier is dynamic, changing every 8 days. This frequent change ensures that even if a malicious node gains access, it cannot impersonate a legitimate node for long.

2.1Illustration:

Consider we have three nodes in our Spark Cluster: "Node1," "Node2," and "Node3.

Step 1: DNA Cryptography to Convert CombinedBinarykey (Hostname, ipaddress, macaddress) to DNA Sequences

"Node1combinedkey" becomes "AGCTTAGG."

"Node2combinedkey" becomes "TGCAGCTT."

"Node3combinedkey" becomes "CTTAGGTC."

Step 2: Mapping to Binary Codes 3. Convert DNA Sequences to Binary:

"AGCTTAGG" becomes "00100011001101000100101000111000."

"TGCAGCTT" becomes "10111000100110100110001011011000."

"CTTAGGTC" becomes "11001011110000100100111001101000."

Step 3: Confusion to Hackers, Convert Binary Codes to Hexadecimal:

"00100011001101000100101000111000" becomes "23A51438."

"10111000100110100110001011011000" becomes "B923468D."

"11001011110000100100111001101000" becomes "C9F12468."

Step 4: Continuous 8-Digit Hostname 5. Initial Hostnames:

"Node1" is assigned the unique key "23A51438."

"Node2" is assigned the unique key "B923468D."

"Node3" is assigned the unique key "C9F12468."

Step 5: Dynamic Key Rotation. Every 8 days, the keys are rotated to the next consecutive 8 digits

Steps	Description	Mac address	IP address	Hostname
	Node Data	5c:ea:1d:99:15:67	192.168.1.11	sathwik
1	Combined Data	5c:ea:1d:99:15:67192.168.1.11sathwik		
2	Binary Form	0011010101100011011001010110000100110001011001000 0111001001110010011000100110101001101100011011100 1100010011100100110010001100010011011000111000001 1000100110001001100010111001101100000101110100011 0 100001110111011010010110 1011		
3	DNA Form	ATCCCGATCGCCCGACATACCGCAATGCATGCATA CATCCATCGATCTATACATGCATAGATACATCGATG AATACATACATACCTATCGACCTCACGGACTCTCGG CCGGT		
4	Mapping (A=0,C=1,G=2, T=3)	0311120312111201030112100321032103010311031203130 3010321030203010312032003010301030113031201131012 20131312211223		
5	Decimal to Hexa(unique key)	8479499BF7DAAA012C0BB09069106B5BDFB839F2BA2 A1B08835773A1C685AB4E09EC7076F3BDFAEBEF54D8 1F9517		

Table 2.1 Generation of unique key through DNA Cryptography

3. DISCUSSION OF RESULTS:

Analysis and comparison were conducted between the conventional and proposed mode, assessing various parameters such as node access. The experimentation utilized an Intel i5 processor with 16GB main memory and another setup with an i5 processor featuring 8GB of memory. The implementation of these algorithms was carried out using VSCode 1.60.0

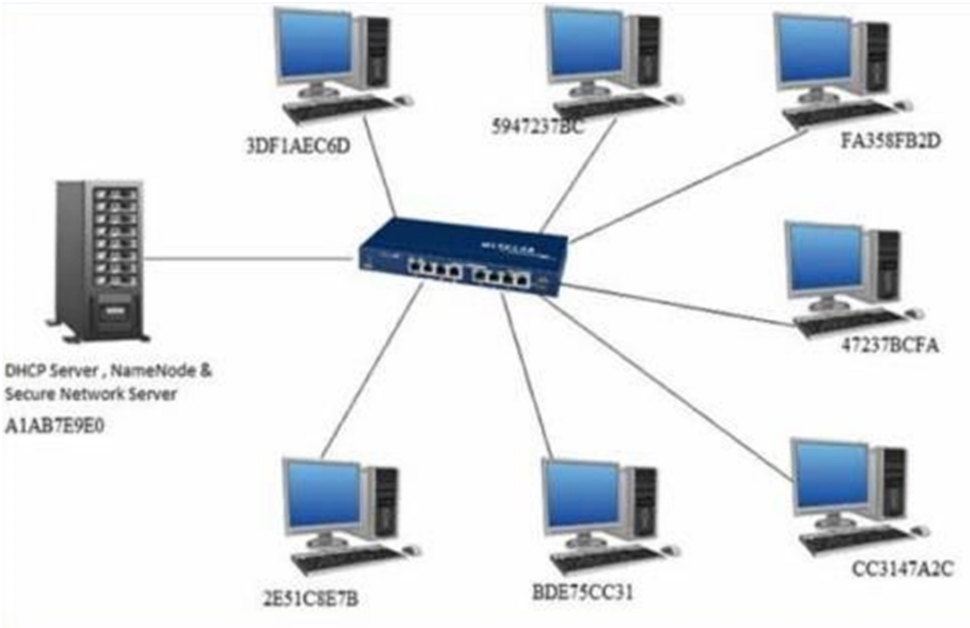

Figure 3.1 Proposed Spark Cluster by Dynamic Hostnames

The experiment successfully demonstrates the generation of dynamic and ever- changing unique keys. The periodic rotation of the keys every 8 days ensures that even if a malicious node gains access to one key, it becomes obsolete quickly. By continuously changing the hostname to a unique key, the system becomes less predictable, reducing the risk of unauthorized access.

	Existing	Proposed (Secure Layer)
Nodes	8	8
Successfully attempt access	6	0

Table 3.2 Nodes and Its Access Information

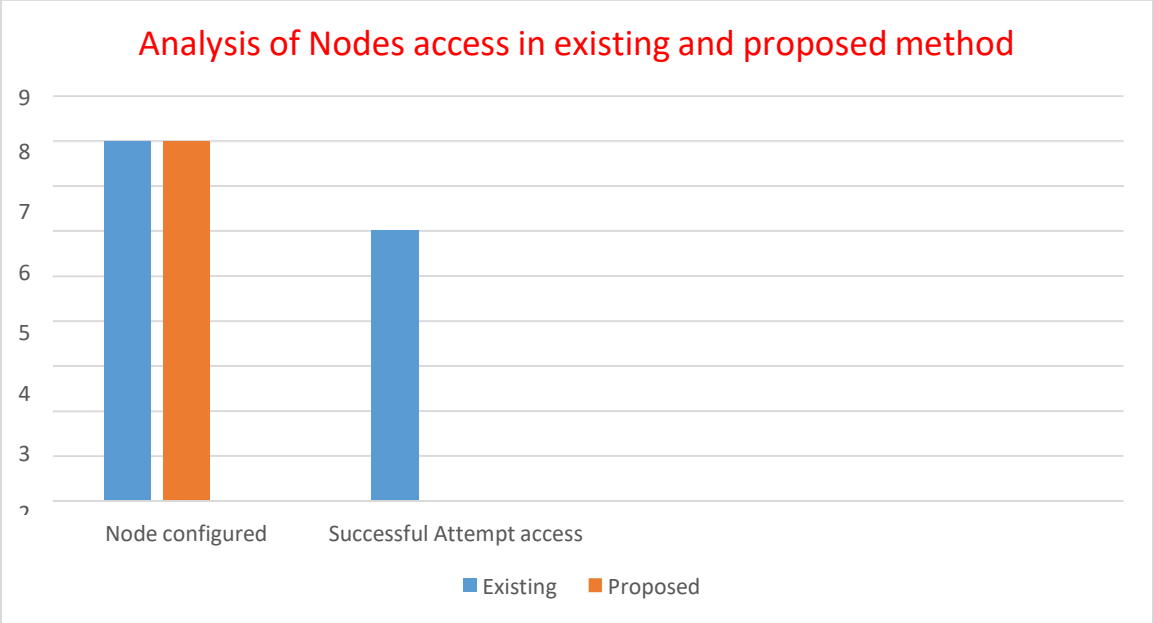

Figure 3.3 Analysis of Nodes access in existing and proposed method

Figure 3.2 compares the node access control mechanisms between the existing method and the proposed method in the context of Spark Cluster management. The analysis focuses on the uniqueness and generation of hostnames, showcasing the enhanced security measures implemented in the proposed method to restrict unauthorized access.

4. SUMMARY:

Experiment Results

Duplicate Hostname Detection: The system successfully detected cases where a new node attempted to enter the network with a hostname that was already in use by an existing node. This detection mechanism prevented the startup of Spark daemons on the new node.

Prompt alert: When a hostname conflict was identified, the system generated an alert. This notification informed the administrator or user about the duplication issue, ensuring that they were aware of the problem.

Avoid Daemons starting for faulty nodes: The system successfully avoided service disruption by refusing to start Spark daemons on nodes with duplicate hostnames. This ensured that existing services on active nodes continued to operate without any interruptions.

5. CONCLUSION:

The experiments demonstrated the system's effectiveness in maintaining a robust and secure network environment by preventing the startup of Spark daemons on nodes with duplicate hostnames.

The experiments demonstrated that the system's ability to block the startup of Spark daemons on nodes with duplicate hostnames

This functionality not only ensured the uniqueness of node identifiers but also contributed significantly to the stability, consistency, and security of the entire Spark cluster.

As a result, the system proved capable of handling new node entries while safeguarding the integrity and reliability of the distributed computing environment.

6. ACKNOWLEDGEMENT:

It is our privilege and pleasure to express my profound sense of respect, gratitude and indebtedness to our guide Dr J Balaraju, Assistant Professor, Department of Computer Science, Anurag University, for his guidance, discussion, encouragement and valuable advice throughout the dissertation work. His motivation in the field of Big Data has made us to overcome all hardships during the course of study and successful completion of the project.

7. REFERENCES:

Mohammed, Hossain, & Parvez. (2015). Design and Implementation of a Secure Campus Network. International Journal of Emerging Technology and Advanced Engineering, 5(7).

Balaraju, J., & Prasada Rao, P. V. R. D. (2020b). Investigation and Finding A DNA Cryptography Layer for Securing Data in Hadoop Cluster. Int. J. Advance Soft Compu. Appl, 12, 3.

Sajisha, K. S., & Mathew, S. (2017). An encryption based on DNA cryptography and steganography. 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA), 162-167. doi:10.1109/ICECA.2017.8212786

Alabady.(2009).Design and Implementation of a Network Security ModelforCooperative Network.International Arab Journal of Technology.

Roy, M. (2020). Data Security Techniques Based on DNA Encryption. In Proceedings of International Ethical Hacking Conference 2019.eHaCON 2019. Advances in Intelligent Systems and Computing (vol. 1065). Springer.doi:1