Science and Technology Journal Vol. 12 Issue: 2 June 2024 ISSN: 2321-3388

Shadowed and Redundant Rules Removal in the Cloud Firewall Policy: A Modified Tree Rule Firewall Approach

Dhwani Hakani[1] and Palvinder Singh Mann[2]
[1]Research Scholar, Gujarat Technological University, Ahmedabad, Gujarat, India
[2]Professor, Department of Computer Science & IT, Central University of Jammu, J&K-181143
E-Mail: [1]Adf_dhwani@gtu.edu.in, [2]palvinder.csit@cujammu.ac.in

Abstract—Firewalls are essential for security and are used to secure the majority of private networks. A firewall's goal is to examine every incoming and outgoing data before granting permission. One common type of conventional firewall is the rule-based firewall. But when it pertains to task performance, traditional listed-rule firewalls are limited, and they don't operate well on some networks with extremely big firewall rule sets. This study proposes a model firewall architecture called "Tree-Rule Firewall," which has benefits and functions well on large-scale networks like "cloud." In order to improve cloud network security, this study suggests an improved tree firewall that eliminates shadowing and redundant rules. Initially, this effort creates a tree rule. The proposed revised tree rule firewall effectively locates the shadow rules while avoiding the creation of redundant rules. Next, a cloud-based test was conducted on an altered Tree-Rule firewall that controls firewall rules. It is demonstrated that increased network security and quicker processing are offered by the modified Tree-Rule firewall. Large networks, such as cloud networks, are easier to build using a modified Tree-Rule firewall since it effectively eliminates shadow and redundant rules.

Keywords: Security, Tree rules, Firewall, redundant, shadowed rule

INTRODUCTION

Firewalls are currently commonly used on the Internet to protect network devices from hostile or undesired traffic [1], which could jeopardize the services' security, integrity, and accessibility. Security policies are implemented by firewalls as a series of rules, each of which consists of an action plus a condition that are defined across a few network header fields. Firewalls use the first matched approach to compare incoming packets to rule conditions gradually until an appropriate rule is found [2]. At this juncture, the packet is either allowed or denied based on the proper action. Firewall rules are defined by network parameters, destination Internet Protocol (IP) addresses and corresponding port numbers [3]. A firewall serves as a barrier between a trusted internal network and untrusted external networks, like the internet. Its primary function is to monitor and control incoming and outgoing network traffic based on predetermined security rules. Firewalls can be hardware, software, or a combination of both.

Here’s a breakdown of how firewalls work:

Packet Filtering: Firewalls inspect packets of data as they pass through. They compare information like source and destination addresses, ports, and packet types against a set of rules. If a packet matches a rule (such as being from a trusted IP address or using a permitted protocol), it's allowed through. Otherwise, it's blocked.

https://doi.org/10.22232/stj.2024.12.02.16

Stateful Inspection: This type of firewall not only examines individual packets but also tracks the state of active connections. It makes decisions based on the context of the traffic, allowing it to better distinguish legitimate packets from potential threats.

Proxy Service: Some firewalls act as intermediaries between clients and servers. Instead of allowing direct connections, they intercept requests from clients and make the requests on their behalf. This adds an extra layer of security by hiding internal network details from external sources.

Application Layer Inspection: Also known as deep packet inspection, this technique involves analyzing the contents of packets at the application layer. Firewalls can identify and block specific application protocols or even specific content within packets, providing granular control over network traffic.

Virtual Private Network (VPN) Support: Firewalls often include VPN capabilities, allowing remote users to securely access the internal network over the internet. VPNs encrypt traffic between the user's device and the firewall, protecting it from eavesdropping and tampering.

Firewalls play a crucial role in network security by enforcing security policies, preventing unauthorized access, and minimizing the risk of cyber attacks. They're a fundamental component of any organization's cyber security infrastructure. If the traffic satisfies the requirements of the filtering rule, the rule takes effect; if not, the subsequent rule in the order sequence takes effect, and so on. The sample rules for firewalls are shown in Table 1.

Table 1: Firewall Rules Sample.

Rule Id	Protocol	SrcIP	SrcPort	Dest IP	Dest Port	Action
R0	TCP	172.168.1.2	Any	123.45.12.36	25	Allow
R1	TCP	10.15.3.42	Any	192.108.1.16	80	Allow
R2	UDP	9.4.5.62	Any	192.108.1.17	25	Deny
R3	TCP	10.15.3.41	Any	3.7.12.78	25	Allow
R4	TCP	178.16.45.12	Any	3.4.12.4	80	Deny
R5	TCP	3.5.12.48	Any	3.5.7.56	25	Allow

Cloud computing is one technology that is incredibly available, private, and versatile. It provides cutting-edge online information exchange services. Adopting cloud computing requires careful consideration of safety concerns [4]. In order to protect the data centre from various attacks, a cloud-based firewall is necessary. High security can be ensured by adhering to the pertinent policies that were developed by a qualified administrator. A cloud gate and a regular firewall are identical save for the fact that the latter is stored on a cloud platform. In essence, cloud-based firewalls are just regular firewalls with an online firewall installed to create a virtual barrier against undesirable network traffic.[5]When it comes to traffic filtering, a virtual router is an application that works similarly as a hardware firewall.In a cloud context, the firewall can be set up as a service or appliance. The cloud-based firewall approach is depicted in Figure 1.

Rule disagreements in the context of firewalls fall into two categories: those pertaining to security and speed. Firewall rule conflicts can occur when two or more rules contradict each other or when the order of rules leads to unintended consequences. Here are some common types of conflicts:

Rule Priority: Firewalls typically process rules in order, from top to bottom. If a packet matches multiple rules, the firewall applies the action of the first matching rule it encounters and ignores the rest. Therefore, the order of rules is crucial Conflicts arise when rules with broader criteria precede more specific rules, leading to unintended access or denial [6].

Allow vs. Deny: Conflicts can arise between rules that allow certain traffic and rules that deny similar traffic. For example, if there's a rule allowing traffic from a specific IP address range but another rule denying traffic from the same range, it creates ambiguity about whether the traffic should be permitted or blocked.

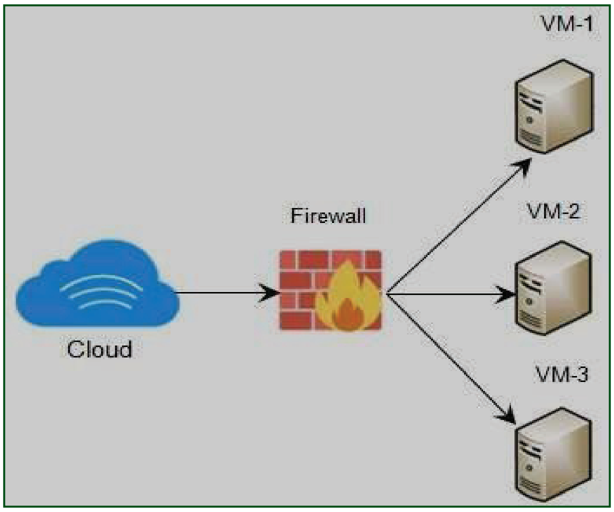


Fig. 1: Firewall Cloud Model.

Conflicting Port Rules: If there are conflicting rules regarding the same port or service, it can lead to confusion about whether the traffic should be allowed or blocked. For instance, one rule might permit traffic on a specific port while another rule denies traffic on the same port.

Inbound vs. Outbound Traffic: Firewalls often have separate rules for inbound and outbound traffic. Conflicts can arise when there are inconsistencies between inbound and outbound rules, leading to unexpected behavior.

Rule Overlaps: Sometimes, rules may overlap in their criteria, causing conflicts in how traffic is handled. For example, if there are two rules allowing traffic from overlapping IP address ranges, it's unclear which rule should take precedence.

To address firewall rule conflicts, it's essential to: Review and prioritize rules carefully to ensure they're logically ordered.

Regularly audit and update firewall rules to remove redundancies and conflicting rules.

Test firewall configurations thoroughly to identify and resolve conflicts before deploying them in production environments.

Document firewall rules and their intended purposes to provide clarity for administrators and troubleshooting purposes.

By proactively managing firewall rules and addressing conflicts, organizations can maintain effective network security and minimize the risk of misconfigurations leading to security breaches.

Moreover, having redundant rules slows down a firewall's processing speed. The reason for this is that they duplicate other rules and eat up processing time of the firewall. Shadowing and superfluous rules must therefore be eliminated from a rule list in order to speed up a firewall's performance.

The improved tree rule firewall suggested in this study would improve cloud network security by getting rid of unnecessary and shadowing rules. The relevant firewall policy's tree rule firewall is initially created by this effort. Next, the suggested improved tree rule firewall effectively locates the shadow rules without generating redundant rules. The main aim of this study's paper is:

- Tree rules firewall is initially built in this work. The initial firewall policy has been correspondingly simplified.
- A sizable number of firewall rules to identify and eliminate the shadowing and repetitive rules. Cloudsim is used to implement and assess the tree rule firewall.

Firewall Tree Rule

[7] He *et al.*'s A firewall tree rule, also known as a hierarchical rule or rule tree, is a concept used in firewall management to organize and structure firewall rules in a hierarchical manner. Instead of having a flat list of rules, where each rule is evaluated independently,[8,9] a firewall tree rule organizes rules into a tree-like structure, allowing for more granular control and efficient management of network traffic.

In a firewall tree rule setup, rules are grouped into categories or nodes, with each node representing a specific level of the hierarchy. Each node can contain sub-nodes or individual rules, further refining the criteria.[10]

For example, consider a firewall tree rule for controlling access to a web server:

At the root level, there might be a node for "Incoming Traffic."

Under "Incoming Traffic," there could be nodes for "HTTP Traffic" and "HTTPS Traffic."

Under "HTTP Traffic," there might be nodes for "Allowed IPs" and "Blocked IPs."

Under "Allowed IPs," there could be specific IP addresses or ranges allowed to access the web server.

By organizing firewall rules in this hierarchical manner, administrators can easily visualize and manage complex rule sets. It also allows for more efficient rule evaluation, as the firewall can quickly navigate the tree structure to determine the appropriate action for incoming or outgoing traffic.[11]

Firewall tree rules are particularly beneficial in large and complex network environments where there are numerous rules governing different aspects of network traffic.[12] They help streamline rule management, improve security posture, and facilitate troubleshooting and auditing processes.

Cloud Firewall

A virtual firewall, as proposed by Jekese *et al*. [13], allows for per-virtual machine network security maintenance, network traffic rule setting, and enhanced virtual environment safety. A private cloud is built using open-source software, and firewall rules are managed using the Tree-Rule approach. With this method, packets are tree-like filtered based on attributes such protocols and IP addresses. Furthermore, the rate that packets are screened and processed has significantly increased in order to keep the virtual firewalls from overloading in this specific scenario.

According to Dezhabad *et al*. [14], the firewall in the cloud should have adaptive auto-scalability. With software installed, this method divides incoming traffic among multiple virtualized firewalls located in a single pool. The goal is to determine, in many time steps, the total number of virtualized firewalls needed, taking into account the volume of traffic and the percentage of requests that reach each firewall.

Bagheri *et al*. [15] describe a method for moving rules from a central barrier to decentralized micro firewalls in a cloud/cloudlets architecture. Rearranging traffic channels during rule migration is required by the solution in order to preserve the network's overall defense policy.

An inspection-based method was developed by Praise *et al*. [16] to thwart malicious behavior by confirming the message signature for incoming communication. It combines the capabilities of reinforcement learning with rapid pattern recognition in parallel to arrive at the best answer as quickly as feasible. The message signature is processed concurrently by the RL approach while learning its environment.

DEFINITIONS

Examples are provided in this section to clarify the meanings of duplicate and shadowing firewall rules.

Redundant Rule

A redundant firewall rule is a rule within a firewall's configuration that duplicates the functionality of another rule or is rendered unnecessary due to the presence of other rules. Redundant rules can arise due to various reasons, such as oversight during rule creation, changes in network topology or policies, or the accumulation of rules over time without proper management.

Having redundant rules in a firewall configuration can lead to several issues:

Reduced Performance: Redundant rules increase the processing overhead on the firewall, as it needs to evaluate each packet against multiple rules unnecessarily. This can degradefirewallperformanceandpotentiallyimpact network throughput. Complexity: Redundant rules contribute to the complexity of the firewall configuration, making it more challenging to understand and manage. This complexity can increase the likelihood of misconfigurations or errors, which could compromise security.

Ambiguity: Redundant rules can introduce ambiguity into the firewall's behavior, leading to unexpected outcomes or conflicts between rules. This ambiguity makes it difficult to predict how the firewall will handle specific types of traffic.

Security Risks: Redundant rules may inadvertently create loopholes or bypasses in the firewall's security policies, leaving the network vulnerable to attacks or unauthorized access. To address redundant firewall rules, network administrators should regularly review and audit the firewall configuration to identify and remove any unnecessary or overlapping rules. This process involves analyzing the rule set, consolidating duplicate rules, and ensuring that each rule serves a specific purpose aligned with the organization's security requirements. Additionally, implementing proper change management processes can help prevent the proliferation of redundant rules in the future.

Shadowing Rule

The rules Ra and Rb only partially match if at least a single value in Ra is not comparable to the corresponding field in Rb.

$$\forall x: \quad R_a(x) \cap R_b(x) \neq \emptyset \ and \ R_a(action) \cap R_b(action) = \emptyset$$

$$Where\ x \in \{protocol, SrcIP, DestIP, SrcPort, DestIP\}$$

$$If\ R_a(proto) \cap R_b(proto) \neq \emptyset\ and\ R_a(IPaddr) \cap R_b(IPaddr) \neq \emptyset \quad {}_a(port) \cap R_b(port) \neq \emptyset\ and\ R_a(action) \cap R_b(action) = \emptyset\ then\ R_a\ is\ shadowed\ by\ R_b$$

When the identical component of rules Ra is a part of rule Rb but their behaviors are different, this is known as shadowing. Since the rule forbids a shadow rule for always having repercussions, shadowing is a serious problem. Consequently, authorized traffic might be forbidden, and vice versa. Consequently, shadowv rules must be located in order for the administrator to move or remove the shadowing rule and fix the problem.

Modified Tree Rule Firewall

This section describes the suggested modified tree trule firewall. The MTRFcloud workflow is depicted in Figure 2.

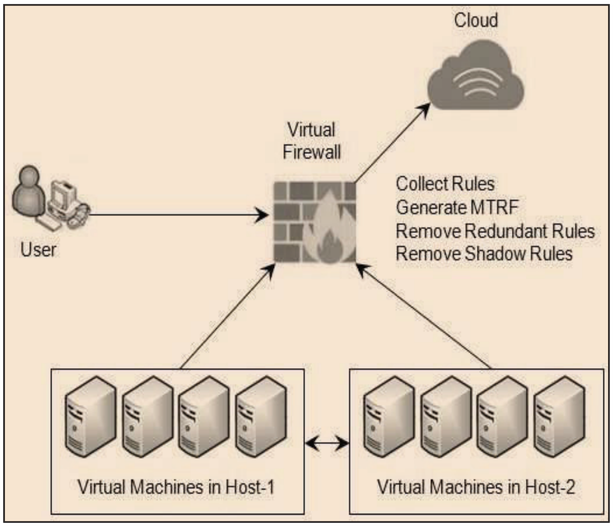


Fig. 2: Modified Tree Firewall Cloud Workflow.

Instead of presenting the rules statically or manually, a new virtualized firewall does it dynamically. The firewall is called a firewall with stateful inspection because it can monitor a flow's connection status in addition to filtering internet traffic utilizing IP addresses, communication protocols, and ports. This firewall will first read the attribute data from the packet then contrast it to the data stored in its root nodes. After then, the firewall will look at the packet's other attributes one after the other, narrowing its search to relevant nodes at the right levels. As a result, a specific action will be taken to quickly determine the traffic. The updated tree rule firewall is produced by means of algorithm-1. The algorithm creates tree rules based on a list of firewalls as input. The firewall branch rule initially has the first rule added to it. Based on the node's index, the remaining criteria are then inserted. The node in this instance indicates the fields (procedure type, destination and source IP, and port) in firewall rules.

Working example for tree rule generation

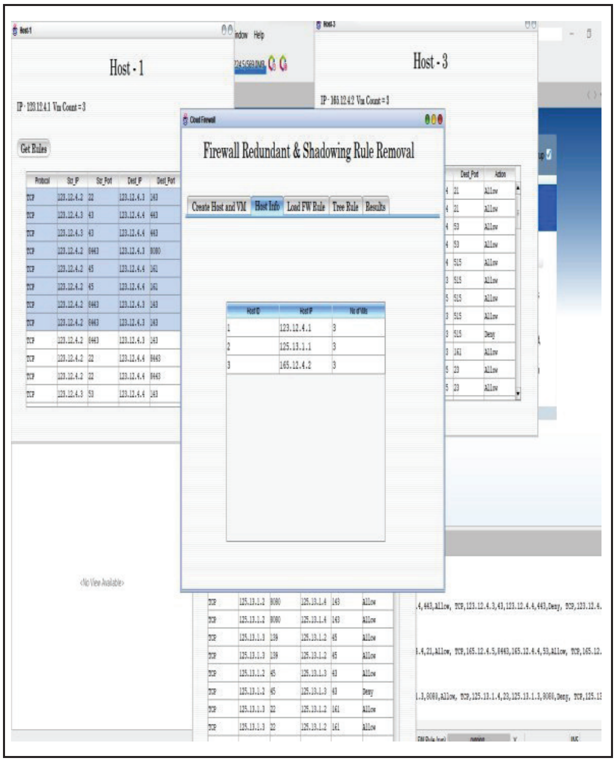
Consider the list of firewall rules

TCP	123.12.4.2 22	123.12.4.3 143	Allow
TCP	123.12.4.3 43	123.12.4.4 443	Allow
TCP	123.12.4.3 43	123.12.4.4 443	Deny
TCP	123.12.4.2 8443	123.12.4.3 8080	Allow
TCP	123.12.4.2 45	123.12.4.4 161	Allow
TCP	123.12.4.2 45	123.12.4.4 161	Allow
TCP	123.12.4.2 8443	123.12.4.3 143	Allow
TCP	123.12.4.2 8443	123.12.4.3 143	Allow

The first rule {TCP 123.12.4.222 123.12.4.3 143 Allow} is inserted into TR. Then, the for-loop is started for adding the remaining rule

Algorithm-1: Firewall Rule Tree Generation

```
currentNode = root
    for criterion in rule.criteria:
        if currentNode has child node matching criterion:
            currentNode = child node matching criterion
        else:
            newNode = create new node with criterion
            currentNode.add(newNode)
            currentNode = newNode
    currentNode.addRule(rule)
    optimizeTree(root)
return root
```



Host - 1

IP - 123.12.4.1 Vm Count = 3

Protocol	Scr_IP	Scr_Port	Dest_IP	Dest_Port
TCP	123.12.4.2	22	123.12.4.3	143
TCP	123.12.4.3	43	123.12.4.4	443
TCP	123.12.4.3	43	123.12.4.4	443
TCP	123.12.4.2	8443	123.12.4.3	8080
TCP	123.12.4.2	45	123.12.4.4	161
TCP	123.12.4.2	45	123.12.4.4	161
TCP	123.12.4.2	8443	123.12.4.3	143
TCP	123.12.4.2	8443	123.12.4.3	143
TCP	123.12.4.2	8443	123.12.4.3	143
TCP	123.12.4.2	22	123.12.4.4	8443
TCP	123.12.4.2	22	123.12.4.4	8443
TCP	123.12.4.3	53	123.12.4.4	143

Host - 3

IP - 165.12.4.2 Vm Count = 3

Dest_Port	Action
21	Allow
21	Allow
53	Allow
53	Allow
515	Allow
515	Allow
515	Allow
515	Allow
515	Deny
161	Allow
23	Allow
23	Allow

Cloud Firewall

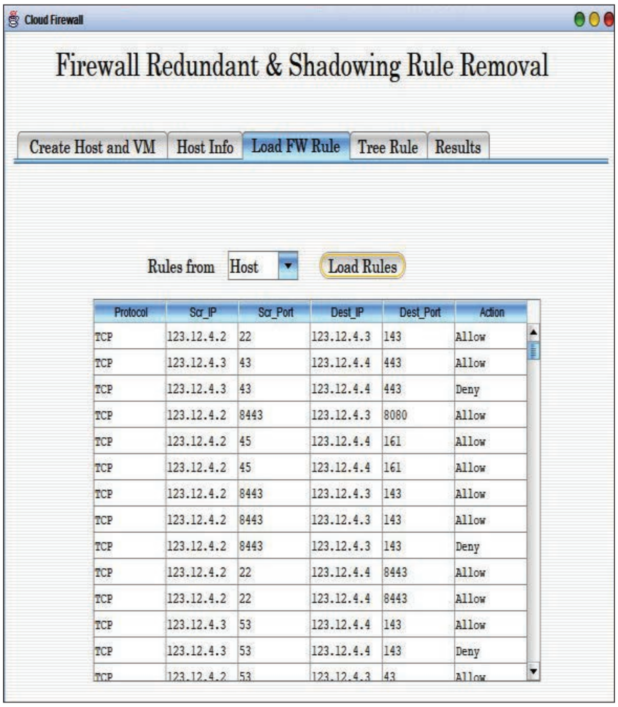
Firewall Redundant & Shadowing Rule Removal

Host ID	Host IP	No of VMs
1	123.12.4.1	3
2	125.13.1.1	3
3	165.12.4.2	3

TCP	125.13.1.2	8080	125.13.1.4	143	Allow
TCP	125.13.1.2	8080	125.13.1.4	143	Allow
TCP	125.13.1.3	139	125.13.1.2	45	Allow
TCP	125.13.1.3	139	125.13.1.2	45	Allow
TCP	125.13.1.2	45	125.13.1.3	43	Allow
TCP	125.13.1.2	45	125.13.1.3	43	Deny
TCP	125.13.1.3	22	125.13.1.2	161	Allow
TCP	125.13.1.3	22	125.13.1.2	161	Allow

```
.4,443,Allow, TCP,123.12.4.3,43,123.12.4.4,443,Deny, TCP,123.12.4.
.4,21,Allow, TCP,165.12.4.5,8443,165.12.4.4,53,Allow, TCP,165.12.
1.3,8080,Allow, TCP,125.13.1.4,23,125.13.1.3,8080,Deny, TCP,125.13
```

Fig. 3: Host Rule Information.



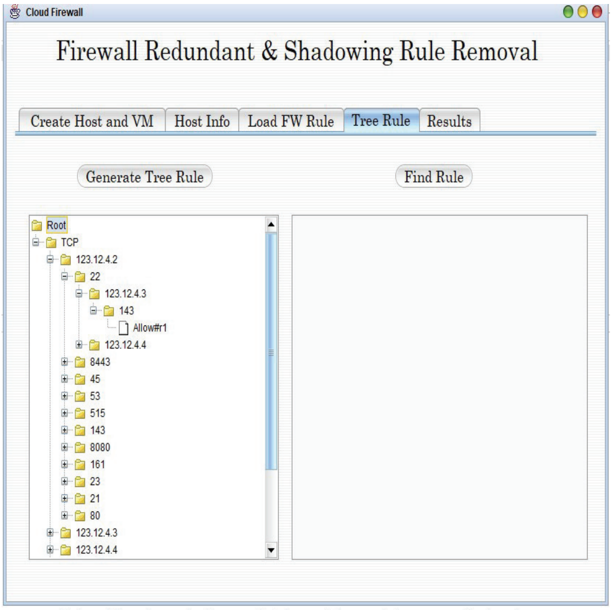
Cloud Firewall

Firewall Redundant & Shadowing Rule Removal

Rules from Host

Protocol	Scr_IP	Scr_Port	Dest_IP	Dest_Port	Action
TCP	123.12.4.2	22	123.12.4.3	143	Allow
TCP	123.12.4.3	43	123.12.4.4	443	Allow
TCP	123.12.4.3	43	123.12.4.4	443	Deny
TCP	123.12.4.2	8443	123.12.4.3	8080	Allow
TCP	123.12.4.2	45	123.12.4.4	161	Allow
TCP	123.12.4.2	45	123.12.4.4	161	Allow
TCP	123.12.4.2	8443	123.12.4.3	143	Allow
TCP	123.12.4.2	8443	123.12.4.3	143	Allow
TCP	123.12.4.2	8443	123.12.4.3	143	Deny
TCP	123.12.4.2	22	123.12.4.4	8443	Allow
TCP	123.12.4.2	22	123.12.4.4	8443	Allow
TCP	123.12.4.3	53	123.12.4.4	143	Allow
TCP	123.12.4.3	53	123.12.4.4	143	Deny
TCP	123.12.4.2	53	123.12.4.3	43	Allow

Fig. 4: Firewall Rules Loaded.



Cloud Firewall

Firewall Redundant & Shadowing Rule Removal

```
Root
  TCP
    123.12.4.2
      22
        123.12.4.3
          143
            Allow#r1
        123.12.4.4
      8443
      45
      53
      515
      143
      8080
      161
      23
      21
      80
    123.12.4.3
    123.12.4.4
```

Fig. 5: Modified Tree Generation.

A firewall rule shouldn't change when a redundant rule is removed. Implementation and Results

Redundant Rules are removed

TCP,123.12.4.2,45,123.12.4.4,161,Allow

TCP,123.12.4.2,8443,123.12.4.3,143,Allow

TCP,123.12.4.2,22,123.12.4.4,8443,Allow

TCP,123.12.4.2,53,123.12.4.3,43,Allow

TCP,123.12.4.2,53,123.12.4.3,22,Allow

TCP,123.12.4.2,53,123.12.4.3,45,Allow

TCP,123.12.4.3,21,123.12.4.2,22,Allow

TCP,123.12.4.2,45,123.12.4.4,443,Allow

TCP,123.12.4.3,143,123.12.4.2,161,Allow

TCP,123.12.4.4,8080,123.12.4.2,43,Allow

Shadowed Rules

TCP,123.12.4.2,8443,123.12.4.3,143,Allow

TCP,123.12.4.2,8443,123.12.4.3,143,Deny

TCP,123.12.4.2,53,123.12.4.3,43,Allow

TCP,123.12.4.2,53,123.12.4.3,43,Deny

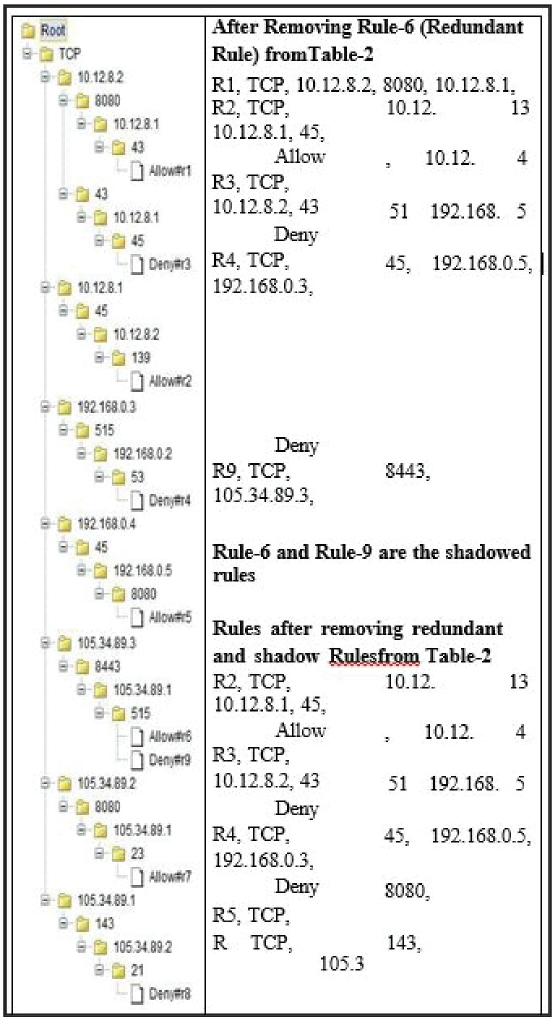
TCP,123.12.4.2,53,123.12.4.3,45,Allow

TCP,123.12.4.2,53,123.12.4.3,45,Deny

Table 2 shows the sample rules collected from VMs.

Table 2: Sample Rules From VMs.

Rule Id	Protocol	SrcIP	SrcPort	Dest IP	Dest Port	Action
R1	TCP	10.12.8.2	8080	10.12.8.1	43	Allow
R2	TCP	10.12.8.1	45	10.12.8.2	139	Allow
R3	TCP	10.12.8.2	43	10.12.8.1	45	Deny
R4	TCP	192.168.03	515	192.168.0.2	53	Deny
R5	TCP	192.168.0.4	45	192.168.0.5	8080	Allow
R6	TCP	192.168.0.4	45	192.168.0.5	8080	Allow
R7	TCP	105.34.89.3	8443	105.34.89.1	515	Allow
R8	TCP	105.34.89.2	8080	105.34.89.1	23	Allow
R9	TCP	105.34.89.1	143	105.34.89.2	21	Deny
R10	TCP	105.34.89.3	8443	105.34.89.1	515	Deny



After Removing Rule-6 (Redundant Rule) fromTable-2

R1, TCP, 10.12.8.2, 8080, 10.12.8.1,
R2, TCP, 10.12. 13
10.12.8.1, 45,
Allow , 10.12. 4
R3, TCP,
10.12.8.2, 43 51 192.168. 5
Deny
R4, TCP, 45, 192.168.0.5,
192.168.0.3,

Deny
R9, TCP, 8443,
105.34.89.3,

Rule-6 and Rule-9 are the shadowed rules

Rules after removing redundant and shadow Rulesfrom Table-2

R2, TCP, 10.12. 13
10.12.8.1, 45,
Allow , 10.12. 4
R3, TCP,
10.12.8.2, 43 51 192.168. 5
Deny
R4, TCP, 45, 192.168.0.5,
192.168.0.3,
Deny 8080,
R5, TCP,
R TCP, 143,
105.3

R6, TCP, 105.34.89.3, 8443, 105.34.89.1, 515,
Allow
R7, TCP, 105.34.89.2, 8080, 105.34.89.1, 23,
Allow
R8, TCP, 105.34.89.1, 143, 105.34.89.2, 21,

Fig. 6: Tree Rules and Results.

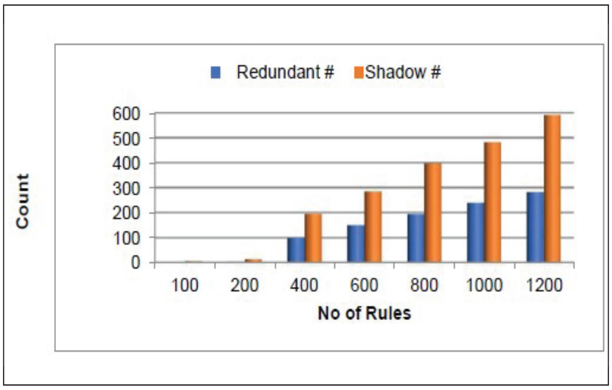


Fig. 7: No of Rules Vs Redundant and Shadow Rules.

Table 3: Number of Redundant and Shadow Rule.

Rules	Redundant	Shadowed
100	3	5
400	100	194
600	151	285
1000	241	485

Table 4 shows the time comparison for different numbers of rules.

Table 4: Time Comparison.

Rule	Tree time for Generation in ms	Processing in ms
100	113	181
400	404	550
600	616	751
1000	1258	3931

Figure 8 shows the execution time of tree generation and processing time.

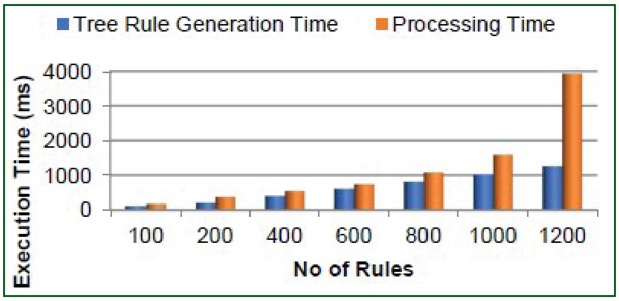


Fig. 8: Execution Time for Different Numbers of Rules.

The total processing time of MTRFcloud is compared with an adaptive cross-domain firewall (ACD) [17] and a double decision tree (DDT) [18]. Figure 9 shows the total processing time comparison of different rules. From that results, the time percentage between ACD and proposed is 32.78% and DDT and proposed is 14.75%.

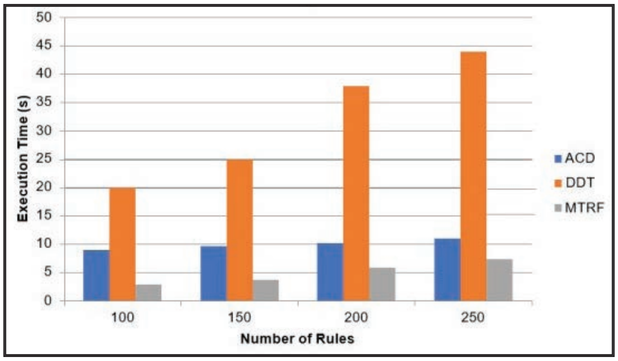


Fig. 9: Total Processing Time Comparison.

CONCLUSION AND FUTURE WORK

This study proposed a modified Tree-Rule firewall to remove shadow and unnecessary rules. The transmitting choice for incoming traffic based on tree policy will follow the tree formation before deciding on the traffic sooner in the MTRFcloud, which employs rules in a tree-based architecture. Since MTRFcloud was extensively tested in a cloud environment, it works best there. Additionally, it was seen to make decisions on packet forwarding quickly. However, the experiment results show that the processing times for generating tree rules and identifying shadowing and redundant rules in a firewall are reasonable in a real-world scenario. Based on those findings, the time percentage difference between the proposed and ACD is 32.78%, and the proposed and DDT is 14.75%.Future work describes removing intra and inter firewall anomalies

REFERENCES

Liu, A. X., Khakpour, A. R., Hulst, J. W., Ge, Z., Pei, D., & Wang, J. (2017). Firewall fingerprinting and denial of firewalling attacks. IEEE Transactions on information forensics and security, 12(7), 1699-1712.

Cheminod, M., Durante, L., Seno, L., & Valenzano, A. (2021). An Algorithm for Security Policy Migration in Multiple Firewall Networks. In ITASEC (pp. 344-359).

Jabal, A. A., Davari, M., Bertino, E., Makaya, C., Calo, S., Verma, D., *et al.* (2019). Methods and tools for policy analysis. ACM Computing Surveys (CSUR), 51(6), 1-35.

Ullrich, J., Cropper, J., Frühwirt, P., & Weippl, E. (2016). The role and security of firewalls in cyber-physical cloud computing. EURASIP Journal on Information Security, 2016(1), 1-20

Toumi, H., Fagroud, F. Z., Zakouni, A., & Talea, M. (2019). Implementing Hy-IDS, mobiles agents and virtual firewall to enhance the security in IaaS Cloud. Procedia Computer Science, 160, 819-824.

Voronkov, A., Iwaya, L. H., Martucci, L. A., & Lindskog, S. (2017). Systematic literature review on usability of firewall configuration. ACM Computing Surveys (CSUR), 50(6), 1-35.

He, X., Chomsiri, T., Nanda, P., & Tan, Z. (2014). Improving cloud network security using the Tree-Rule firewall. Future generation computer systems, 30, 116-126.

Chomsiri, T., He, X., Nanda, P., & Tan, Z.(2016). Hybrid tree-rule firewall for high speed data transmission. IEEE transactions on cloud computing, 8(4), 1237-1249.

Chomsiri, T., He, X., Nanda, P., & Tan, Z. (2014, September). A stateful mechanism for the tree-rule firewall. In 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications (pp. 122-129). IEEE.

Suresh, N., & Bai, B. M. (2016). Predictive Modelling of Tree Rule Firewall for the Efficient Packet Filtering. International Journal of Computer Science and Information Security, 14(10), 189.

Trabelsi, Z., Masud, M. M., & Ghoudi, K.(2015). Statistical dynamic splay tree filters towards multilevel firewall packet filtering enhancement. Computers & Security, 53, 109-131.

Trabelsi, Z., Zeidan, S., Shuaib, K., & Salah, K. (2018). Improved session table architecture for denial of stateful firewall attacks. IEEE Access, 6, 35528-35543.

Jekese, G., & Hwata, C., "Virtual Firewall Security on Virtual Machines in Cloud Environmen", International Journal of Scientific and Engineering Research, 6(2), 2015

Dezhabad, N., & Sharifian, S. (2018). Learning-based dynamic scalable load-balanced firewall as a service in network function-virtualized cloud computing environments. The Journal of Supercomputing, 74, 3329-3358.

Bagheri, S., & Shameli-Sendi, A. (2020). Dynamic firewall decomposition and composition in the cloud. IEEE Transactions on Information Forensics and Security, 15, 3526- 3539.

Praise, J. J., Raj, R. J. S., & Benifa, J. B. (2020). Development of Reinforcement Learning and Pattern Matching (RLPM) Based Firewall for Secured Cloud Infrastructure. Wireless Personal Communications, 115, 993-1018.

Kadam, P. R., & Bhusari, V. K. (2014) Redundancy removal of rules with reordering them to increase the firewall optimization. International Journal of Research in Engineering and Technology, 3(10), 317-321.

Lin, Z., & Yao, Z. (2022). Firewall Anomaly Detection Based on Double Decision Tree. Symmetry, 14(12), 2668.