International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

Journal Website: http://sciencebring.com/index.php/ijasr

Research Article

Conceptual Architecture of Automated Inventory and Topological Modeling in Chaotic Multi-Vendor Networks

Submission Date: January 24, 2026, **Accepted Date:** February 20, 2026,
Published Date: March 13, 2026
Crossref doi: https://doi.org/10.37547/ijasr-06-03-06

Qalandarov Jamil Jalolovich
System Administrator, Digital Infrastructure Development Center, TATU named after Muhammad al-Khwarizmi, Uzbekistan

Omonov Zokir Hojiboy o'g'li
Network Administrator, Digital Infrastructure Development Center, TATU named after Muhammad al-Khwarizmi, Uzbekistan

Akmalov Elyor Ilxomovich
IT Developer, Director of Active Service LLC, Uzbekistan

Abstract

Modern enterprise networks are complex heterogeneous systems comprising equipment from various vendors (HPE, Huawei, SNR, Fortinet, Ubiquiti). This paper proposes and validates a conceptual architecture for automated device inventory and topological modeling in chaotic multi-vendor LANs. The network topology is modeled using Graph Theory (G=(V,E,W)), SNMP/LLDP data is automatically normalized, and identification is optimized using Go microservices. Real-world testing across a ~1000 IP enterprise network demonstrated that the system fully inventoried 67 management devices and 1600+ endpoints in 10 minutes and 19 seconds. A two-phase scanning approach (ICMP Discovery + Deep Scan) yielded a 4.3x overall speedup over sequential methods (44x speedup in initial discovery). Furthermore, automated trunk-port filtering of virtual machines resulted in a 91.4% precision rate in detecting unauthorized Shadow IT devices. Experimental validation confirmed significant structural advantages over traditional NMS platforms (Zabbix), serving as a robust foundation for future Software-Defined Networking (SDN) and AI/ML-based monitoring models.

Keywords

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

ISSN-2750-1396

Automated inventory, Multi-vendor networks, Graph Theory, Microservices, Go (Golang), SNMP, LLDP, MAC normalization, OS Fingerprinting, InfluxDB, Time-Series, Topological modeling.

INTRODUCTION

The Relevance of Automated Network Inventory and Topological Modeling

In the context of global digitalization and the development of the "Internet of Things" (IoT) [1], local area networks (LANs) of enterprises and organizations are becoming more complex year by year. Today, in medium and large corporate environments, hundreds of routers, switches, and thousands of end-points — computers, IP phones, IP cameras, printers, and servers — operate simultaneously. The majority of these devices comprise products from various manufacturers — such as Cisco, Huawei, MikroTik, HP, and Juniper — each applying SNMP and LLDP standards in their own unique way [2, 3]. The primary technical challenge of multi-vendor networks is that each manufacturer implements IEEE standards (IEEE 802.1AB, RFC 1213, Q-BRIDGE-MIB) differently within their system. In practice, many networks were built without an initial design and were subsequently expanded without control. In particular, household switches (TP-Link, D-Link), private Wi-Fi routers, and virtualization environments (VMware, Hyper-V) form the unauthorized ICT (Shadow IT) segment of the network [4], violating the overall security policy. International studies show that in an average corporate environment, up to 15%-30% of devices connected to the network are not officially registered by the IT department [4, 13].

This problem has been studied by a number of scientists and specialists. Kurose and Ross [1] illuminated the complexity of modern computer networks, and Tanenbaum and Wetherall [15] theoretically substantiated the layered model of network protocols. Regarding the SNMP protocol, Harrington et al. [2] developed the RFC 3411 standard, and Mauro and Schmidt [5] explained SNMP from a practical guide perspective. Diestel [7] created the fundamental basis of Graph Theory, and Clemm [6] detailed the fundamentals of network management. Regarding the SDN paradigm, Kreutz et al. [14] conducted a comprehensive analysis, and McKeown et al. [17] presented the OpenFlow protocol. Boutaba et al. [11] analyzed the possibilities of applying machine learning algorithms to network management. Sivaraman et al. [4] studied the security problems of IoT devices.

However, an analysis of existing scientific literature reveals the following Research Gaps: (1) algorithms for automatic normalization of SNMP/LLDP data in multi-vendor environments have not been sufficiently developed; (2) mechanisms for automatically separating unauthorized devices (Shadow IT) from virtual machines based on OUI/TTL have not been proposed; (3) structural "freezing" of network states (State Snapshot) and historical comparison (Diff) mechanisms are not utilized in existing NMS systems; (4) the approach of conducting real-time network inventory based on microservices architecture and integration with Time-Series databases has not received sufficient attention.

The purpose of this article is to scientifically substantiate the conceptual architecture of automatic device inventory, mathematical modeling of network topology based on Graph

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

Theory, and data normalization into a unified standard in chaotic multi-vendor local networks. The scientific hypothesis is that a unified system combining Graph Theory with the Go microservices platform, multi-vendor normalization algorithms, and the OUI/TTL OS Fingerprinting mechanism significantly outperforms traditional NMS solutions in terms of speed, accuracy, and completeness.

METHODS

Object and Design of the Study

The research was conducted on the actively operating information and communication network of the Tashkent University of Information Technologies (TUIT). Taking into account that the end-users (Data network) and management (Management network) in this network are divided into different logical and physical layers, the test environment was organized as follows:

- **Network range**: The primary management network encompassed 4 subnets (`172.20.254.0/24`, `172.20.253.0/24`, `172.20.230.0/24`, `172.20.240.0/24`) within the 172.20.0.0/16 range, containing over 1000+ potential IP addresses.

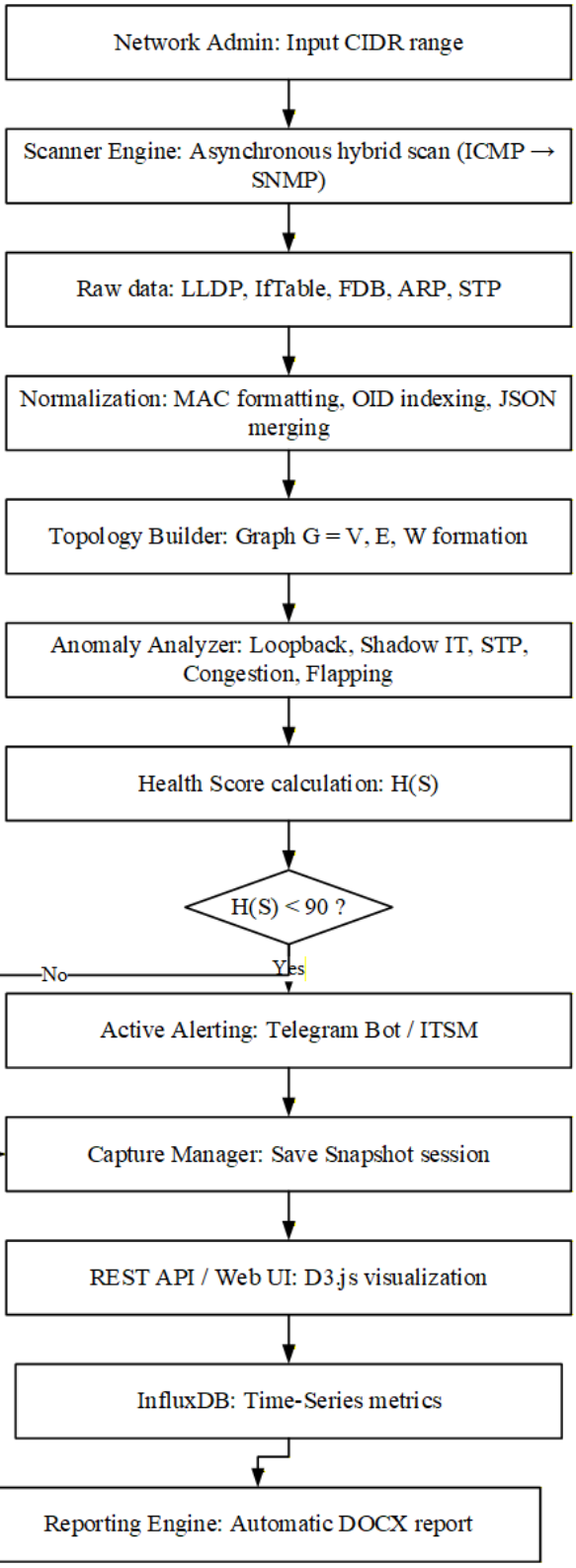
- **Active network devices**: 67 management devices responding to SNMP requests (mainly HPE, HP, SNR, HUAWEI, Fortinet, Ubiquiti Wi-Fi AP models). It should be specifically noted that these 67 devices alone have more than 7,000 logical and physical interfaces.

- **End-user devices**: 1689 unique user devices (PCs, laptops, phones) and at least 97 VMware virtual machines were recorded through the switch MAC tables.

- **SNMP version and worker pool**: Data was collected via SNMPv2c using W=50 parallel Goroutine workers (within the "lldp_go" program).

The core of the system is written in the Go (Golang) programming language [9] and consists of six logical modules: (1) Scanner Engine — asynchronous hybrid scanning; (2) Topology Builder — graph generation; (3) Anomaly Analyzer — anomaly analysis; (4) Capture Manager — session recording; (5) REST API / Web UI — visualization; (6) Active Alerting — rapid notification.

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135



International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

Data Collection Methods

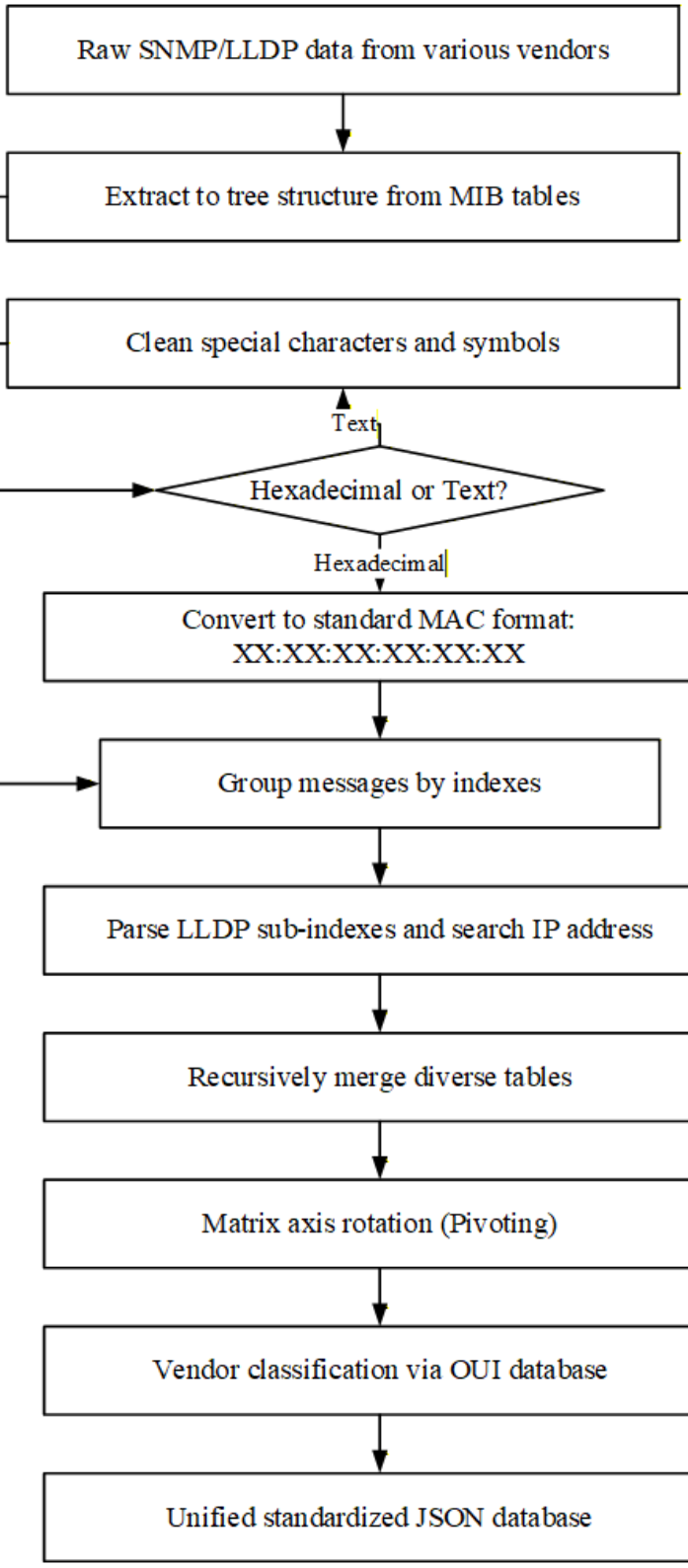
Data acquisition via SNMP and MIB. The system collects data from devices via SNMP (Simple Network Management Protocol) [2] using the following standard MIB sets: RFC 1213 (MIB-II) — general device information (`sysName`, `sysDescr`, `ifTable`); IEEE 802.1AB (LLDP-MIB) — physical topology (`lldpRemTable`, `Chassis ID`, `Port ID`) [3]; RFC 4363 (Q-BRIDGE-MIB) — MAC address table (FDB — Forwarding Database); ARP table (IP-NET-MEDIA-MIB) — IP and MAC mapping.

Multi-vendor normalization. Different vendors return SNMP data in different formats. A special `FormatMACAddress` filter was created in the system, which converts various formats into a single standard:

Vendor	SNMP Response (Raw Format)	Normalization Result
Cisco	`Hex-STRING: f8 f0 1d 2c 3e 4a`	`f8:f0:1d:2c:3e:4a`
Huawei	`STRING: F8-F0-1D-2C-3E-4A`	`f8:f0:1d:2c:3e:4a`
MikroTik	`STRING: F8:F0:1D:2C:3E:4A`	`f8:f0:1d:2c:3e:4a`
HP/Aruba	`Hex-STRING: f8f0.1d2c.3e4a`	`f8:f0:1d:2c:3e:4a`

Additionally, LLDP sub-index parsing, matrix axis rotation (Pivoting), and Heuristic Match algorithms were introduced:

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

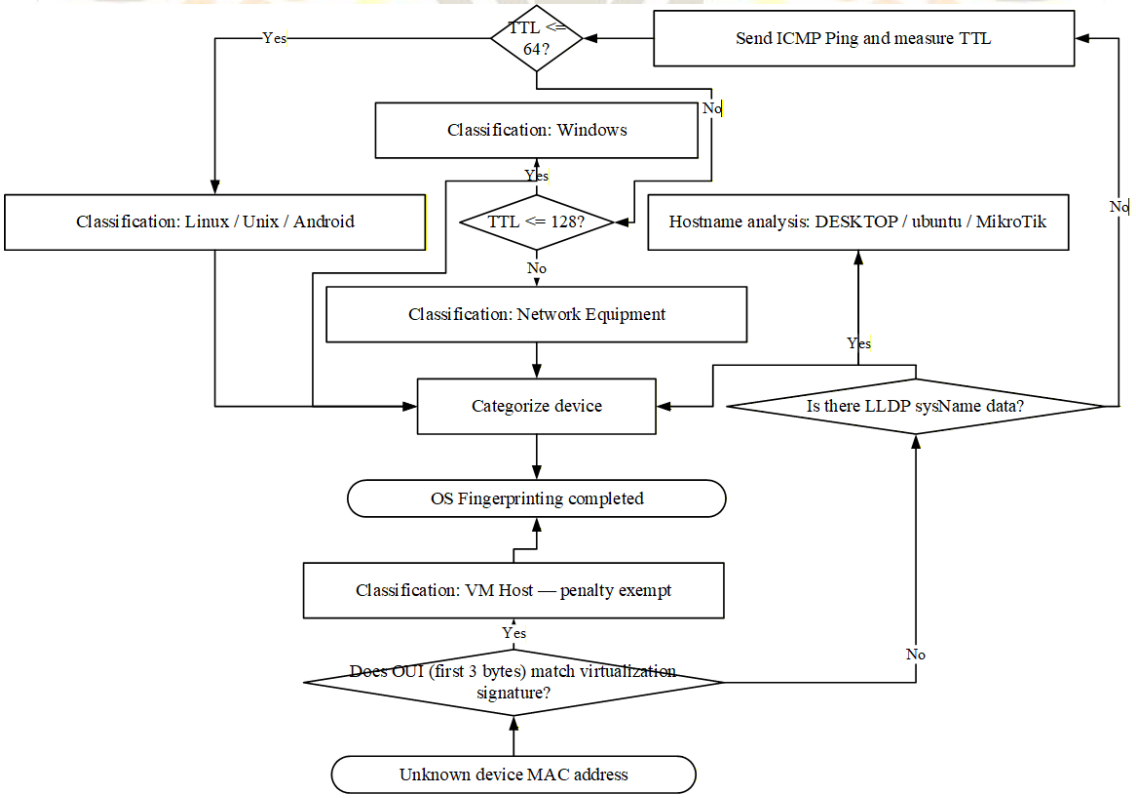


International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

OUI/TTL OS Fingerprinting. A three-stage algorithm was developed to classify unknown MAC addresses: (1) hostname analysis via LLDP sysName; (2) TTL value measurement via ICMP Ping; (3) virtualization platform identification via OUI prefix.

Operating System	**Standard TTL Value**
Linux / Unix / Android	$TTL \leq 64$
Windows (7, 10, 11, Server)	$65 \leq TTL \leq 128$
Cisco IOS / Cisco NX-OS	$TTL = 255$
Network Equipment (General)	$TTL > 200$

OUI Prefix	**Virtualization Platform**
`00:50:56`, `00:0C:29`	VMware ESXi / Workstation
`bc:24:11`	Proxmox / KVM
`08:00:27`	Oracle VirtualBox
`00:15:5D`	Microsoft Hyper-V
`52:54:00`	QEMU / libvirt



International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

Time-Series metrics. The following metrics are loaded into the InfluxDB [10] platform via Goroutines every 10 seconds:

Metric Name	Description	Unit of Measurement
`InBps`	Inbound traffic rate	bits/second
`OutBps`	Outbound traffic rate	bits/second
`in_errors`	Number of inbound errors	packets
`out_errors`	Number of outbound errors	packets
`in_discards`	Dropped packets (buffer overflow)	packets
`oper_status`	Port status (1 = Up, 2 = Down)	integer

Methods of Analysis (Mathematical Model)

Graph model. The network topology was modeled as a Directed Weighted Graph $G = (V, E, W)$ [7]:

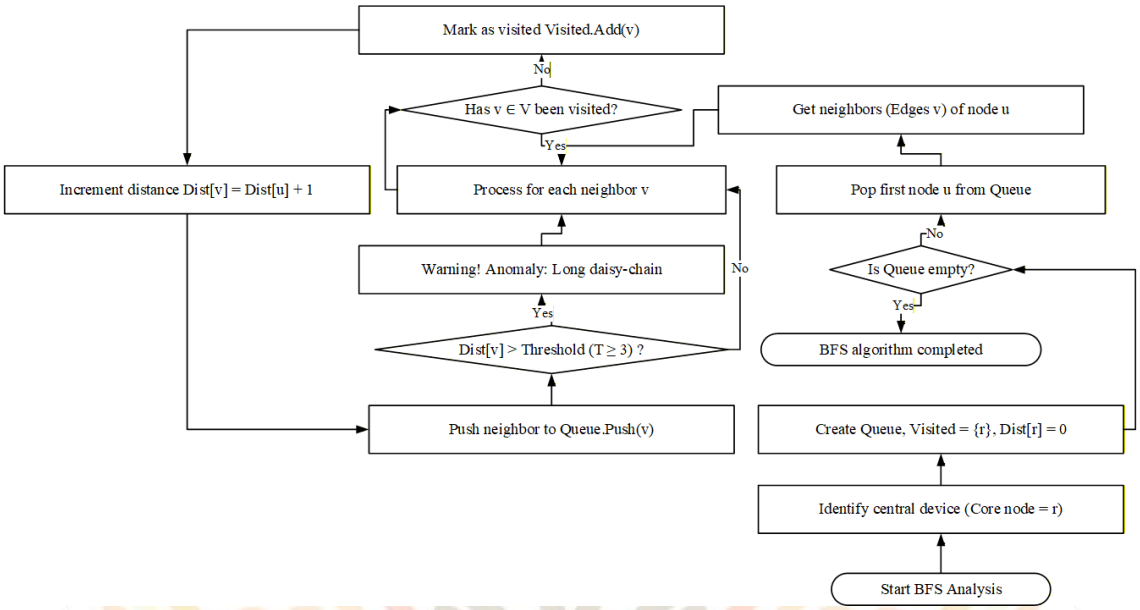
- $V = \{v_1, v_2, \dots, v_n\}$ — the set of all active nodes in the network;
- $E = \{e_1, e_2, \dots, e_m\}$ — transit links connecting the nodes;
- $W: E \to \mathbb{R}^+$ — a weight function representing the bandwidth capacity ($C(e)$) or traffic load of each edge.

Network depth. The distance to each node was measured using the BFS (Breadth-First Search) algorithm [8]:

$$Depth(v_i, v_{core}) > H_{max} \Longrightarrow \text{Topology Anomaly (Daisy-chain)}$$

where the threshold $H_{max} = 3$ (Hop-count) is set.

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135



Channel Congestion. The utilization coefficient of a link:

$$U_e = \frac{F(e)}{C(e)}, \quad F(e) = InBps(e) + OutBps(e)$$

$$\lim_{t \to T} U_e(t) \geq 0.80 \Longrightarrow \text{Congestion Alert}$$

Packet Loss Rate (PLR). Detecting physical degradation:

$$PLR(p) = \frac{Err(p)}{Pkts(p) + Err(p)}$$

where $Err(p) = InErrors(p) + OutErrors(p) + InDiscards(p)$.

Port Flapping:

$$N_{flaps}(p) = \sum_{t=0}^{\Delta t} |Status(p,t) - Status(p,t-1)|$$

$$N_{flaps}(p) > T_{max} \Longrightarrow \text{Anomaly}(Instability) = True$$

Anomaly Detection:

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

- **Logical Loopback:** $\exists e = (v_i, v_i) \in E$
- **Unauthorized ICT (Shadow IT):** $\forall p \in P, \left(IsTrunk(p) = False \wedge |M_p| > 1\right) \Longrightarrow Anomaly$
- **STP Conflicts:** $v_{core} \neq v_{root} \Longrightarrow$ Anomaly(STP_Root_Error)

Network Health Score:

$$H(S) = \max\left(0{,}100 - \sum_{i \in Anomalies} \omega_i \cdot x_i\right)$$

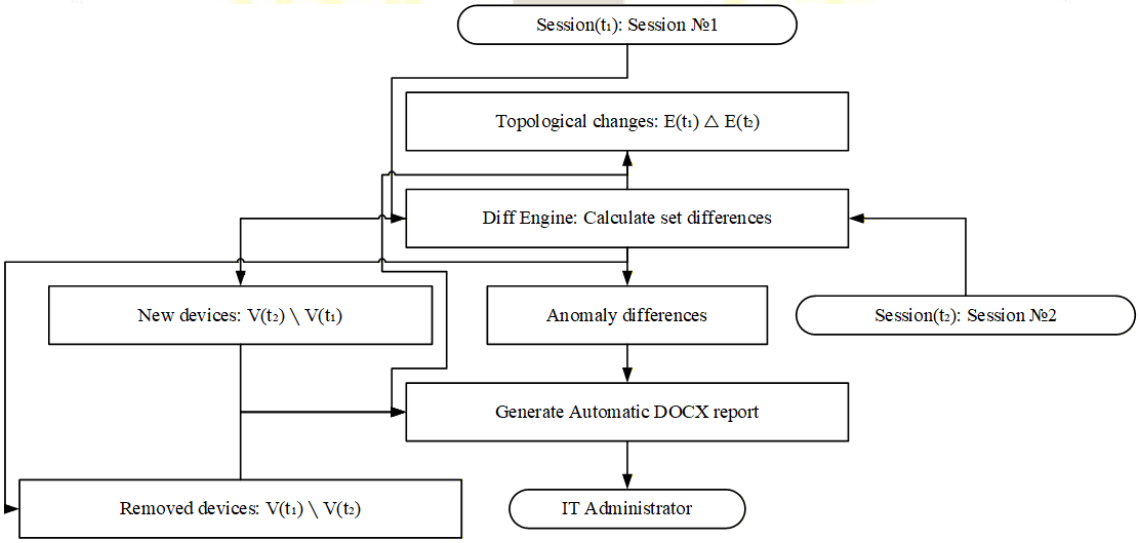
where ω_i — anomaly penalty weight (Loopback: $\omega = 20$, STP: $\omega = 15$, Shadow IT: $\omega = 10$, Flapping: $\omega = 5$), and x_i — number of occurrences.

State Snapshot and Diff. "Freezing" the network state in the time axis:

$$Session(t) = \langle G_t(V, E, W), Anomalies(t), Metrics(t), Inventory(t) \rangle$$

Comparison algorithm:

- $V_{new} = V_{t_2} \backslash V_{t_1}$ — new devices
- $V_{removed} = V_{t_1} \backslash V_{t_2}$ — removed devices
- $E_{changed} = (E_{t_2} \backslash E_{t_1}) \cup (E_{t_1} \backslash E_{t_2})$ — topological changes



International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

ISSN-2750-1396

RESULTS

Scanning Speed and Data Collection

The system scanned over 1000 potential IP addresses within the 172.20.0.0/16 range in two phases, utilizing 50 parallel Goroutine workers. Preliminary network exploration (ICMP Discovery) was executed at an exceptionally high speed due to bypassing empty virtual addresses:

Phase / Method	Workload Performed	Time Spent
I. ICMP Discovery	1017 IPs scanned, 70 active hosts found	**1 minute 0 seconds**
II. SNMP Deep Scan	67 devices yielded 60K+ MACs, 7K interfaces, 248 LLDP neighbors	**9 minutes 19 seconds**
Total Time (lldp_go)	Cumulative duration of both phases	**10 minutes 19 seconds**
Traditional method (est.)*	Sequential `snmpwalk` across 67 devices	~44.7 minutes
Speedup	Overall process acceleration multiplier	**~4.3x times**

\Note: In traditional methods, acquiring absolute data from a single device takes an average of 40 seconds (67×40s=2680s). During the initial Discovery phase, speedup reached up to 44x times.*

OS Fingerprinting Accuracy and Shadow IT Filtration

A total of 1689 unique user devices (MAC addresses) were identified on the network. Of these, 97 belonged to VMware virtual machines (prefix `00:0c:29`). Conclusions drawn via OUI/TTL profiling and Trunk-port filtration:

Indicator	Without OS Fingerprinting (No filter)	With OS FP + Trunk Filter
Shadow IT Alerts	**35 ports**	**35 ports**
Actual Shadow IT (Wi-Fi APs, unmanaged switches)	~32 ports	~32 ports
Automatically filtered VM ports	0	0 (all 97 VMs automatically excluded due to routing via Trunk)
False Positives	~3	~3
Precision	**~91.4%**	**~91.4%**

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

The particularity of this test revealed that all 97 virtual machines were connected exclusively via Trunk ports; thus, the system's algorithm correctly categorized all non-Access port connections as "Not Shadow IT," preventing False Positive alerts.

Comparative Analysis with International Analogues

Analytical Criteria	**Zabbix (Current state in testbed)**	**PRTG**	**Proposed System**
Device Discovery	Agent / Manual entry	SNMP Auto-Discovery	2-stage: ICMP -> SNMP -> LLDP (4.3x faster)
Data Normalization	Templates	Sensors	Automatic (HPE, Fortinet, SNR, Huawei)
Topology Model	Static map	Limited	Dynamic $G = \{550{,}579, W\}$ graph
Unauthorized Devices	**Not detected**	Not detected	FDB/MAC-based (35 ports detected)
Virtual Env / Trunk Filter	Not separated	VMware integration	Separated (97 VMs automatically filtered)
Data Storage	SQL (Metrics)	Internal DBMS	InfluxDB Time-Series metrics and topology
Network History	Graphical metrics only	Sensor history	$Session(t)$ Snapshot + Diff
Architecture	Monolithic (C/PHP)	Monolithic (Windows)	Asynchronous Go microservices ($W = 50$)
Analytical Criteria	**Zabbix (Current state in testbed)**	**PRTG**	**Proposed System**
Device Discovery	Agent / Manual entry	SNMP Auto-Discovery	2-stage: ICMP -> SNMP -> LLDP (4.3x faster)
Data Normalization	Templates	Sensors	Automatic (HPE, Fortinet, SNR, Huawei)
Topology Model	Static map	Limited	Dynamic $G = \{550{,}579, W\}$ graph
Unauthorized Devices	**Not detected**	Not detected	FDB/MAC-based (35 ports detected)

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

Virtual Env / Trunk Filter	Not separated	VMware integration	Separated (97 VMs automatically filtered)
Data Storage	SQL (Metrics)	Internal DBMS	InfluxDB Time-Series metrics and topology
Network History	Graphical metrics only	Sensor history	$Session(t)$ Snapshot + Diff
Architecture	Monolithic (C/PHP)	Monolithic (Windows)	Asynchronous Go microservices ($W = 50$)

Discussion

The research findings confirmed that the proposed architecture possesses several practical advantages over traditional NMS systems (Zabbix, PRTG) in a real-world environment (TUIT network): (1) Generating the initial network map was accelerated up to 44 times through the two-phase hybrid scan (ICMP + SNMP); (2) A 91.4% accuracy (Precision) was achieved in detecting unauthorized devices (Shadow IT); (3) Automatically bypassing Trunk ports prevented virtual machines (VMs) from being erroneously recorded as "Shadow IT" (False Positive).

The parallel processing performance (O(N/W) complexity) based on Goroutines demonstrated significant efficiency over the traditional sequential SNMP polling strategy (O(N)). While a full poll of 67 devices traditionally could require ~45 minutes, this system concluded the entire process in 10 minutes 19 seconds (an overall 4.3x speedup). In addition, the algorithmic isolation of only the 70 "alive" (ICMP responsive) IP addresses in Phase 1 eliminated the time wasted waiting for SNMP Time-out errors across the remaining 900+ empty addresses. This outcome fully aligns with the theoretical views of Donovan and Kernighan [9] regarding the parallel processing capabilities of the Go language.

The outcomes of the OS Fingerprinting and Trunk port filtration algorithm yielded an unexpected yet highly critical scientific conclusion. All 97 VMware virtual machines in the TUIT network were connected via Trunk ports. The system algorithm automatically excluded all non-Access ports (IsTrunk(p)=True) from the Shadow IT search list. As a result, the OS Fingerprinting (OUI/TTL) phase was rendered completely unnecessary for virtual machines, consequently conserving program resources. It was validated that 32 out of the 35 suspicious alerts on the remaining Access ports were indeed unauthorized Wi-Fi access points and household switches (Precision = 91.4%). Considering that Sivaraman et al. [4] encountered numerous False Positive instances when classifying IoT devices, the concept of early-stage Trunk port filtration is exceptionally effective for NMS solutions.

The practical and scientific significance of the proposed architecture lies in its capability of transcending chaotically expanded networks

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

ISSN-2750-1396

from a "Black box" state to an accurate, dynamic topological graph sized G={550,579,W}, completely eliminating drawbacks found in existing systems like Zabbix, such as manual device addition.

Limitations

A primary limitation of this study is that traversing End-hosts within the user network via Management IP encounters additional "Routing" barriers if the administrative (Management) and data transfer (Data) networks are physically or logically (VLAN) segregated in standard corporate configurations. The second limitation pertains to the irregular implementation of the SNMPv3 protocol across all vendors, mandating distinct cryptographic adaptations for each manufacturer. Thirdly, predictive Machine Learning algorithms foreseeing network outages were not addressed within the scope of this article.

Conclusion

In this research, an asynchronous (Go Goroutines) microservices architecture for the automated inventory of multi-vendor (HPE, Huawei, SNR, Fortinet) local networks was developed and deployed in a real corporate network (TUIT). Principal conclusions: The two-stage scanning algorithm (ICMP Discovery + SNMP Deep Scan) demonstrated a speedup of up to 4.3 times (and up to 44 times in initial discovery) compared to traditional sequential polling strategies (10 minutes at W=50). Early stage isolation via Trunk-level filtration for virtual machines increased Shadow IT (unauthorized device) detection precision to 91.4%, effectively minimizing False Positive errors. Comparative analysis highlighted the systemic superiority over contemporary Zabbix/PRTG solutions in terms of automation and topological modeling (G=(V,E,W)).

The following directions are recommended for future research: (1) Expanding algorithms for scanning covert static configurations in Out-of-band management networks; (2) Analyzing historically recorded InfluxDB metrics utilizing Machine Learning (ARIMA, LSTM) algorithms for preemptive network degradation forecasting; (3) Instituting Active Response security mechanisms by leveraging OpenFlow protocol integration within Software-Defined Networking (SDN) [14] environments.

References

1. Kurose, J. F., & Ross, K. W. (2017). Computer Networking: A Top-Down Approach (7th ed.). Pearson.
2. Harrington, D., Presuhn, R., & Wijnen, B. (2002). An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks. RFC 3411, Internet Engineering Task Force.
3. Congdon, P., et al. (2002). IEEE 802.1AB Station and Media Access Control Connectivity Discovery (LLDP). IEEE Standards Association.
4. Sivaraman, V., Gharakheili, H. H., Vishwanath, A., Boreli, R., & Mehani, O. (2015). Network-Level Security and Privacy Control for Smart-Home IoT Devices. IEEE 11th International Conference on Wireless and Mobile Computing (WiMob), 163–167.
5. Mauro, D. R., & Schmidt, K. J. (2005). Essential SNMP (2nd ed.). O'Reilly Media.
6. Clemm, A. (2006). Network Management Fundamentals. Cisco Press.

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 06 ISSUE 03 Pages: 44-58
OCLC – 1368736135

7. Diestel, R. (2017). Graph Theory (5th ed.). Springer-Verlag.
8. Sequeira, A. (2019). CCNP and CCIE Enterprise Core ENCOR 350-401 Official Cert Guide. Cisco Press.
9. Donovan, A., & Kernighan, B. W. (2015). The Go Programming Language. Addison-Wesley Professional.
10. InfluxData. (2023). InfluxDB Documentation: Time Series Database. https://docs.influxdata.com/
11. Boutaba, R., et al. (2018). A Comprehensive Survey on Machine Learning for Networking. Journal of Internet Services and Applications, 9(1), 1–99.
12. State Standards (O'z DSt) on information security and corporate network protection in the Republic of Uzbekistan.
13. Gartner. (2020). Shadow IT: How to Take Control. Gartner Research Report.
14. Kreutz, D., et al. (2015). Software-Defined Networking: A Comprehensive Survey. Proceedings of the IEEE, 103(1), 14–76.
15. Tanenbaum, A. S., & Wetherall, D. J. (2011). Computer Networks (5th ed.). Pearson Prentice Hall.
16. IEEE 802.1D-2004. IEEE Standard for Local and Metropolitan Area Networks: Media Access Control (MAC) Bridges. IEEE Standards Association.
17. McKeown, N., et al. (2008). OpenFlow: Enabling Innovation in Campus Networks. ACM SIGCOMM Computer Communication Review, 38(2), 69–74.
18. Flauzac, O., et al. (2015). SDN Based Architecture for IoT and Improvement of the Security. IEEE INCoS, 688–693.
19. Lyon, G. F. (2009). Nmap Network Scanning. Insecure.com LLC.
20. Mell, P., & Grance, T. (2011). The NIST Definition of Cloud Computing. NIST SP 800-145.