INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD)
© 2026 IJNRD | Volume 11, Issue 6, June 2026 | ISSN: 2456-4184 | IJNRD.ORG

REAL-TIME ATTACK MONITORING AND LOG ANALYSIS USING COWRIE HONEYPOT AND ELK STACK

[1]A M Chandrashekar, [2]M Shivakumar
[1]Associate Professor, [2]PG Student
[1,2]Department of Computer Science and Engineering SJCE, JSS Science And Technology University Mysuru, Karnataka, India

Abstract **:** The rapid growth of cyber threats has increased the need for advanced monitoring and threat detection systems in modern network environments. Traditional security mechanisms such as firewalls and intrusion detection systems often fail to provide detailed visibility into attacker behaviour after intrusion attempts. This paper presents a real-time attack monitoring and log analysis framework using the Cowrie Honeypot integrated with the ELK Stack (Elasticsearch, Logstash, and Kibana). Cowrie acts as a medium-interaction SSH and Telnet honeypot that captures attacker activities including brute-force login attempts, executed commands, and malicious file downloads. The generated logs are forwarded using Filebeat to Logstash for processing and then stored in Elasticsearch for indexing and analysis. Kibana provides real-time dashboards and graphical visualization of attack patterns. The proposed system was tested using simulated attacks from Kali Linux tools such as Hydra and Nmap. Experimental results demonstrate the effectiveness of the framework in capturing and visualizing malicious activities in real time. The system provides enhanced visibility into attacker behavior and supports proactive cybersecurity monitoring and threat intelligence.

Keywords-**Cowrie Honeypot, ELK Stack, Elasticsearch, Kibana, Logstash, Cybersecurity, SSH Attack, Real-Time Monitoring, Threat Detection, Honeypot.**

I. INTRODUCTION

Cybersecurity has become one of the most critical concerns in the modern digital world. Organizations rely heavily on interconnected systems, cloud computing, web applications, and remote access technologies, making them vulnerable to various cyber threats such as brute-force attacks, malware infections, ransomware, phishing attacks, and unauthorized access attempts. Attackers continuously develop advanced techniques to exploit vulnerabilities in network infrastructures and applications.

Traditional security solutions including firewalls, antivirus software, and intrusion detection systems (IDS) mainly focus on detecting known attack signatures and blocking malicious traffic. However, these systems often fail to analyse attacker behaviour in depth and provide limited information about post-compromise activities. Additionally, modern attackers use stealth techniques and zero-day exploits that bypass conventional defences.

To overcome these limitations, honeypots have emerged as an effective cybersecurity mechanism. A honeypot is a deliberately vulnerable system designed to attract attackers and record their actions in a controlled environment. Honeypots help researchers and security analysts understand attacker methodologies, attack patterns, exploited vulnerabilities, and malicious intentions without risking production systems.

This paper focuses on implementing a real-time attack monitoring system using the Cowrie Honeypot integrated with the ELK Stack. Cowrie is an open-source SSH and Telnet honeypot that emulates a Linux shell environment and records attacker interactions such as login attempts, commands executed, and file downloads. The ELK Stack consists of Elasticsearch, Logstash, and Kibana, which together provide centralized log collection, processing, storage, and visualization capabilities.

The proposed system enables real-time monitoring and analysis of malicious activities through interactive dashboards and graphical representations. Attack simulations using Kali Linux tools are performed to validate the effectiveness of the framework.

II. PROBLEM STATEMENT

Traditional cybersecurity monitoring systems mainly focus on detecting known attack signatures and blocking malicious traffic. However, they fail to provide detailed visibility into attacker behaviour after successful intrusion attempts. Existing intrusion detection systems generate large volumes of logs, making manual analysis difficult and time-consuming. Furthermore, many organizations lack affordable real-time monitoring systems capable of visualizing attack patterns and suspicious activities effectively.

The absence of centralized log analysis and real-time visualization reduces the ability of security analysts to detect threats proactively. Therefore, there is a need for an integrated and cost-effective solution that can capture attacker activities, process logs centrally, and provide graphical insights into cyberattacks in real time.

III. LITERATURE SURVEY

Several cybersecurity monitoring systems and honeypot technologies have been developed to detect and analyze cyber threats. Traditional security tools such as firewalls, antivirus software, and IDS solutions mainly focus on prevention rather than studying attacker behavior.

Intrusion Detection Systems such as Snort and Suricata monitor network traffic and generate alerts for suspicious activities. However, they are limited in capturing detailed attacker interactions after intrusion attempts. Security Information and Event Management (SIEM) systems such as Splunk and ArcSight provide centralized log analysis but are often expensive and resource-intensive.

Honeypots provide an alternative approach by acting as decoy systems that attract attackers and record malicious activities. Earlier honeypots such as Kippo simulated SSH services and collected login attempts and command histories. However, Kippo lacked real-time visualization and modern logging features.

Cowrie, a successor of Kippo, provides enhanced capabilities including SSH and Telnet emulation, JSON logging, session playback, and malware download tracking. Despite these improvements, Cowrie alone lacks centralized monitoring and advanced visualization capabilities.

The ELK Stack has become a widely adopted open-source solution for centralized logging and real-time analytics. Elasticsearch provides distributed indexing and searching, Logstash processes and enriches logs, and Kibana visualizes data through dashboards.

Several researchers have proposed integrating honeypots with SIEM technologies to improve cybersecurity monitoring. However, many implementations lack real-time visualization, scalability, or detailed attacker behavior analysis. The proposed system addresses these limitations by integrating Cowrie with the ELK Stack for real-time monitoring and visualization.

IV. PROPOSED SYSTEM

The proposed system integrates Cowrie Honeypot with the ELK Stack to create a real-time attack monitoring and analysis framework. The system captures malicious activities performed by attackers and visualizes them using centralized dashboards.
The architecture consists of the following components:

A. Cowrie Honeypot

Cowrie is a medium-interaction honeypot that simulates SSH and Telnet services. It records attacker activities such as:

- Failed and successful login attempts
- Commands executed
- File uploads and downloads
- Session activities
- Malware execution attempts

B. Filebeat

Filebeat is used to monitor Cowrie log files continuously and forward logs to Logstash in real time.

C. Logstash

Logstash processes incoming logs, parses JSON data, extracts useful fields, and enriches the logs for structured analysis.

D. Elasticsearch

Elasticsearch stores processed logs as searchable documents and enables fast querying and indexing.

E. Kibana

Kibana visualizes the logs through interactive dashboards, charts, and graphs, enabling security analysts to monitor attack patterns effectively.

F. Kali Linux Attacker Machine

Kali Linux is used to simulate attacks such as brute-force SSH login attempts, port scanning, and command execution to test the honeypot system.

V. SYSTEM ARCHITECTURE

The system architecture consists of three major layers:

1. Attack Simulation Layer
2. Log Processing Layer
3. Visualization Layer

In the attack simulation layer, attackers interact with the Cowrie Honeypot through SSH or Telnet protocols.
Cowrie records all attacker activities and stores them in JSON log format.

In the log processing layer, Filebeat forwards the logs to Logstash, where they are parsed and enriched before being stored in Elasticsearch.
In the visualization layer, Kibana retrieves indexed data from Elasticsearch and displays it through dashboards and graphical visualizations.
The architecture enables centralized monitoring, real-time analysis, and improved visibility into attacker behavior.

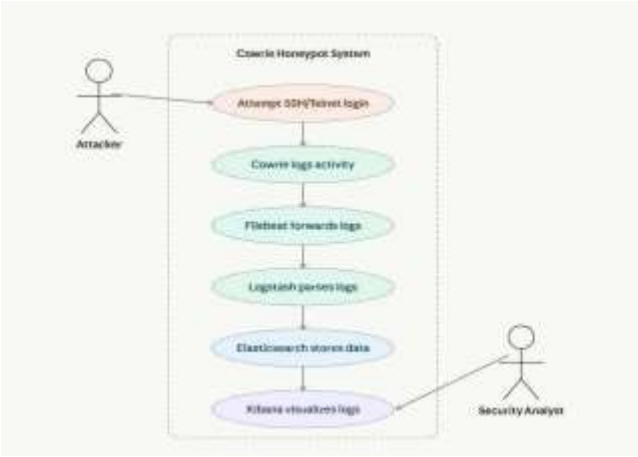

Fig.1. System Workflow of Cowrie Honeypot Integrated with ELK Stack

The proposed system architecture is designed to provide a centralized platform for capturing, processing, and analyzing cyberattack activities in real time. The Cowrie Honeypot acts as the primary component for collecting attacker interactions by emulating SSH and Telnet services. All activities performed by attackers, such as login attempts, command execution, and file transfer operations, are recorded and stored as log files. These logs are continuously monitored by Filebeat and forwarded to Logstash for processing and filtering. The processed data is then indexed and stored in Elasticsearch, enabling efficient storage and fast retrieval of security events. Finally, Kibana visualizes the collected information through interactive dashboards and graphical reports, allowing security analysts to monitor attack patterns and analyze attacker behavior effectively. This architecture ensures seamless data flow between components and supports real-time threat monitoring and cybersecurity analysis.

VI. IMPLEMENTATION METHODOLOGY

A. Virtual Environment Setup

The project was implemented using virtual machines created using Oracle VirtualBox. Three virtual machines were used:

- Ubuntu Server for Cowrie Honeypot
- Ubuntu Server for ELK Stack
- Kali Linux for attack simulation

Static IP addresses were assigned to ensure communication between the systems.

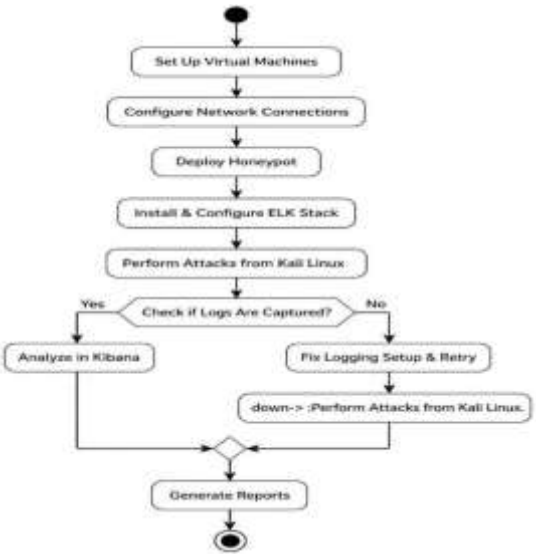

Fig. 2. Activity Diagram of the Proposed Real-Time Monitoring System

B. Installation of Cowrie

Cowrie was installed on Ubuntu Server using Python virtual environments. Necessary dependencies including Python3, Git, and development libraries were installed.

The honeypot was configured to emulate:

- SSH service on port 2222
- Telnet service on port 2223

JSON logging was enabled for centralized monitoring.

Fig. 3. Installation of Cowrie Honeypot from GitHub Repository

C. Installation of ELK Stack

Elasticsearch, Logstash, and Kibana were installed on a dedicated Ubuntu server. Elasticsearch was configured for single-node deployment.

Logstash pipelines were configured to parse Cowrie JSON logs and extract important fields such as:

- Source IP address
- Destination port
- Username
- Commands executed
- Session ID

Elasticsearch was installed and configured to store and index attack logs collected from the Cowrie Honeypot as shown in Fig. 4.

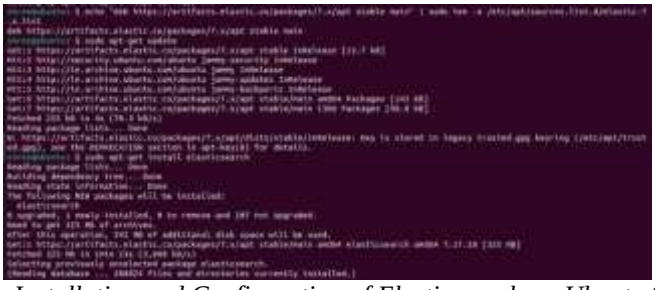

Fig. 4. Installation and Configuration of Elasticsearch on Ubuntu Server

Logstash was configured to parse and process the incoming Cowrie log files before forwarding them to Elasticsearch as illustrated in Fig. 5.

Fig. 5. Installation and Configuration of Logstash for Log Processing

Kibana was installed to visualize and analyze attack logs through real-time dashboards and graphical interfaces as shown in Fig. 6.

Fig. 6. Installation and Configuration of Kibana Dashboard Interface

D. Filebeat Configuration

Filebeat was configured on the Cowrie server to monitor Cowrie log files and forward logs to Logstash using port 5044.

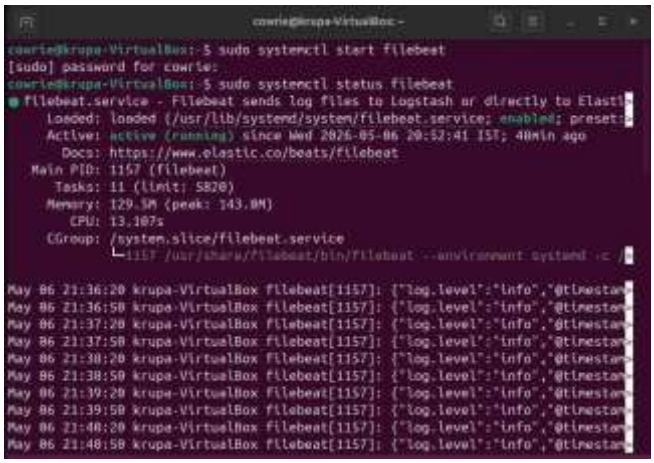

Fig. 7. Filebeat Configuration and Status Verification on Ubuntu Server

E. Kibana Dashboard Setup

Kibana dashboards were created to visualize:

- Number of login attempts
- Top attacker IP addresses
- Most targeted ports
- Frequently executed commands
- Attack timelines

Kibana provides a web-based interface for visualizing and monitoring logs generated by the Cowrie Honeypot, as shown in Fig. 8.

Fig. 8. Kibana Dashboard Home Interface

The incoming attack logs and system activities were visualized in real time using Kibana stream dashboards as illustrated in Fig. 9

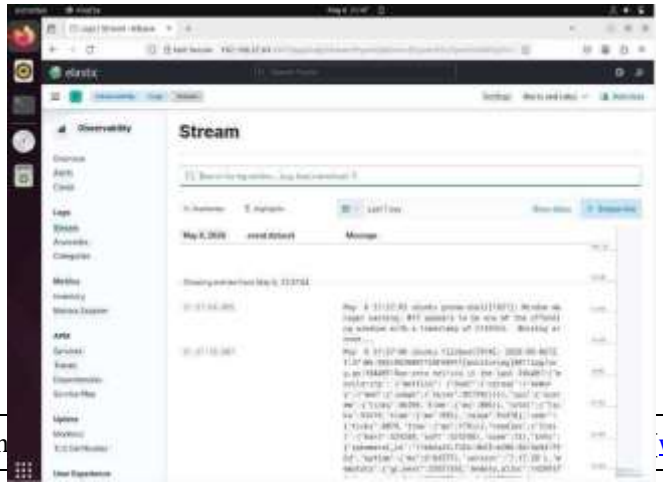

Fig. 9. Real-Time Log Stream Visualization in Kibana

VII. EXPERIMENTAL RESULT

Several attacks were simulated using Kali Linux tools to test the effectiveness of the system.

A. SSH Brute-Force Attack

Hydra was used to perform brute-force SSH attacks against the Cowrie Honeypot. Cowrie successfully captured:

- Username attempts
- Password attempts
- Source IP addresses
- Timestamps

The logs were forwarded to Elasticsearch and visualized in Kibana dashboards.

An SSH-based attack was simulated from the Kali Linux attacker machine to interact with the Cowrie Honeypot and execute Linux commands, as shown in Fig. 10.

Fig. 10. SSH Login and Command Execution on the Cowrie Honeypot

The Cowrie Honeypot successfully recorded SSH session activities, executed commands, and attacker inputs in JSON log format, as illustrated in Fig. 11.

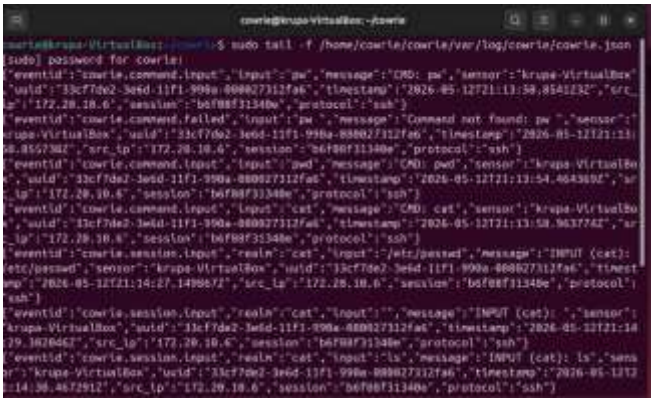

Fig. 11. Cowrie Capturing Real-Time SSH Session and Command Logs

B. Port Scanning

Nmap scans were performed against the honeypot server. The system successfully recorded scanning activities and displayed them in Kibana.

Nmap scanning was performed from the Kali Linux attacker machine to identify open ports and services running on the Cowrie Honeypot, as shown in Fig. 12.

Fig. 12. Nmap Port Scanning Performed on the Cowrie Honeypot

The Cowrie Honeypot successfully recorded SSH connection attempts and session activities generated during the Nmap scan, as illustrated in Fig. 13.

Fig. 13. Cowrie Logs Capturing Port Scanning and SSH Connection Attempts

C. Command Execution Monitoring

After successful login attempts, attackers executed commands such as:

- ls
- pwd
- wget
- uname
- cat /etc/passwd

Cowrie recorded all commands and session activities.

D. Malware Download Detection

The honeypot detected malicious download attempts using wget and curl commands. Download URLs and file names were captured and stored.

E. Visualization Results

Kibana dashboards displayed:

- Top attacker countries
- Most common usernames
- Login attempt frequency
- Timeline-based attack analysis
- Command execution statistics

The system provided real-time visibility into malicious activities.

The collected Cowrie Honeypot logs were indexed in Elasticsearch and visualized through the Kibana Discover dashboard for real-time monitoring and analysis, as shown in Fig. 14.

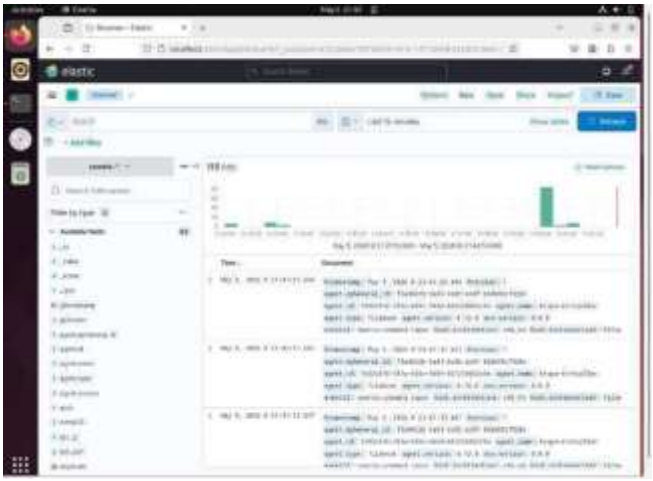

Fig. 14. Visualization of Cowrie Honeypot Logs in Kibana Dashboard

VIII. ADVANTAGES OF THE PROPOSED SYSTEM

The proposed framework offers several advantages:

- Real-time attack monitoring
- Centralized log analysis
- Detailed attacker behavior tracking
- Open-source and cost-effective
- Easy scalability
- Interactive dashboards and visualization
- Improved cybersecurity awareness

- Useful for SOC and threat intelligence teams

IX. LIMITATIONS

Despite its advantages, the system has some limitations:

- Honeypots may be detected by advanced attackers
- High interaction honeypots require additional resources
- Large-scale deployments require higher storage capacity
- False positives may occur during analysis

X. FUTURE ENHANCEMENTS

Future improvements may include:

- AI and Machine Learning integration
- Automated alert generation
- Email and SMS notifications
- Cloud-based deployment
- Integration with threat intelligence platforms
- Multiple distributed honeypots
- Advanced malware analysis

XI. CONCLUSIONS

This paper presented a real-time attack monitoring and log analysis framework using Cowrie Honeypot integrated with the ELK Stack. The system successfully captured and analysed malicious activities such as brute-force attacks, command execution, and malware download attempts. Filebeat, Logstash, Elasticsearch, and Kibana together provided centralized logging and visualization capabilities.

The implementation demonstrated that honeypots combined with SIEM technologies can significantly improve cybersecurity monitoring and attacker behavior analysis. Real-time dashboards enabled better visibility into attack patterns and assisted in proactive threat detection. The proposed system can serve as an effective low-cost solution for cybersecurity research, security operations centres, and educational environments.

REFERENCES

[1] N. Provos, “A Virtual Honeypot Framework,” Proceedings of the 13th USENIX Security Symposium, 2004.
[2] Elastic, “ELK Stack Documentation,” Available: https://www.elastic.co
[3] Cowrie Documentation, Available: https://cowrie.readthedocs.io
[4] M. Nawrocki et al., “Honeypot Systems and Their Applications,” IEEE Access, vol. 4, pp. 102-120, 2020.
[5] R. Bace and P. Mell, “Intrusion Detection Systems,” NIST Special Publication, 2001.
[6] Snort Project Documentation, Available: https://www.snort.org
[7] Suricata IDS Documentation, Available: https://suricata.io
[8] T. Liston, “Honeypots and Network Security,” SANS Institute Reading Room, 2018.
[9] Oracle VirtualBox Documentation, Available: https://www.virtualbox.org
[10] Hydra Tool Documentation, Available: https://github.com/vanhauser-thc/thc-hydra
[11] A. M. Chandrashekhar and K. Raghuveer, “Improvising Intrusion detection precision of ANN based NIDS by incorporating various data Normalization Technique – A Performance Appraisal”, International Journal of Research in Engineering & Advanced Technology(IJREAT), Volume 2, Issue 2, Apr-May, 2014.
[12]A. M Chandrashekhar and K. Raghuveer, “Hard Clustering Vs. Soft Clustering: A Close Contest for Attaining Supremacy in Hybrid NIDS Development”, Proceedings of International Conference on Communication and Computing (ICC - 2014), Elsevier science and Technology Publications.
[13]] Yadunandan Huded, A.M.Chandrashekhar, Sachin Kumar H S, “Advances in Information security risk practices” International Journal of Advanced Research in data mining and Cloud computing (IJARDC), Volume 3, Issue 5 May 2015.
[14]Huda Mirza Saifuddin, A.M.Chandrashekhar, Spoorthi B.S, “Exploration of the ingredients of original security” International Journal of Advanced Research in Computer Science and Applications(IJARCSA), Volume 3, Issue 5, May 2015.
[15]A Chandrashekhar, J Vijay Kumar - Fuzzy min-max neural network-based intrusion detection system,
Proceedings of the International Conference on Nano electronics and communications,2017
[16]Puneeth L Sandal, A. M Chandrashekhar, Prashanth Chillabatte, “Network Security situation awareness system” International Journal of Advanced Research in Information and Communication Engineering(IJARICE), Volume 3, Issue 5, May 2015.
[17]A. M. Chandrashekhar, Jagadish Revapgol, Vinayaka Pattanashetti, “Security Issues of Big Data in Networking”, International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Volume 2, Issue 1, JAN-2016.
[18]A. M. Chandrashekhar and K. Raghuveer, “Fortification of hybrid intrusion detection system using variants of neural networks and support vector machines”, International Journal of Network Security & Its Applications
(IJNSA) ISSN: 0974-9330[online]& 0975- 2307[print]. Vol.5, Number 1, January 2013.
[19]A. M. Chandrashekhar, Sahana K, Yashaswini K,” Securing Cloud Environment using Firewall and VPN”, “International Journal of

Advanced Research in Computer Science and Software Engineering (IJARCSSE), Volume 6, Issue-1, January-2016.
[20]Prashanth G M, A.M.Chandrashekhar, Anjaneya Bulla, "Secured infrastructure for multiple group communication" International Journal of Advanced Research in Information and Communication Engineering (IJARICE), Volume 3, Issue 5, May 2015.
[21]A. M. Chandrashekhar, Arpitha, Nidhishree G, "Efficient data accessibility in cloud with privacy and authenticity using key aggregation cryptosystem", International Journal for Technological research in Engineering (IJTRE), Volume 3, Issue 5, JAN-2016.
[22]] A. M. Chandrashekhar, Hariprasad M, Manjunath A, "The Importance of Big Data Analytics in the Field of Cyber Security", International journal of scientific Research and Development (IJSRD), Volume 3, Issue 11, JAN-2016 pp 439-445.
[23]A. M. Chandrashekhar, Chitra K V, Sandhya Koti, "Security Fundamentals of Internet of Things", International Journal of Research (IJR), Volume 3, Issue no1, JAN-2016.