DEVELOPMENT AND IMPLEMENTATION OF A MICROSOFT WINDOWS LOG ANALYZER

A thesis submitted in partial fulfilment of the requirement for the degree of

Master of Engineering

Department of Computer Science and Engineering
Jadavpur University, Kolkata

By
Suchismita Ghosh
Registration No.: 140747 of 2017-2018

Examination Roll No.: M4CSE19016

Under the Guidance of

Prof. Chandan Mazumdar

Department of Computer Science and Engineering

FACULTY OF ENGINEERING AND TECHNOLOGY

Jadavpur University, Kolkata-700032

India

May, 2019

JADAVPUR UNIVERSITY

FACULTY OF ENGINEERING AND TECHNOLOGY

Certificate of Recommendation

This is to certify that the dissertation entitled "**Development and Implementation of a Microsoft Windows Log Analyzer**" has been satisfactorily carried out by SuchismitaGhosh (University Registration No.: 140747 of 2017-2018, Examination Roll No.: M4CSE19016). It is a bonafide piece of work carried out under my guidance and supervision and be accepted in partial fulfillment of the requirement for the degree of Master of Engineering, Department of Computer Science and Engineering, Faculty of Engineering and Technology, Jadavpur University, Kolkata.

…………………………………………………………………

Prof. ChandanMazumdar (Thesis Supervisor)
Department of Computer Science and Engineering
Jadavpur University, Kolkata-32

Countersigned

……………………………………………………

Prof. MahantapasKundu
Head, Department of Computer Science and Engineering,
Jadavpur University, Kolkata-32.

…………………………………………………….

Prof. ChiranjibBhattacharjee
Dean, Faculty of Engineering and Technology,
Jadavpur University, Kolkata-32.

JADAVPUR UNIVERSITY

FACULTY OF ENGINEERING AND TECHNOLOGY

Certificate of Approval

This is to certify that the thesis entitled “Development and implementation of a Microsoft Windows Log Analyzer” is a bonafide record of work carried out by SuchismitaGhosh in partial fulfilment of the requirements for the award of the degree of Master of Engineering, in Department of Computer Science and engineering, Jadavpur University during the period of July 2018 to May 2019. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose for which it has been submitted.

…………………………………………………………………….

Signature of Examiner

Date:

…………………………………………………………………….

Signature of Supervisor

Date:

Acknowledgement

I am pleased to express my deepest gratitude to my thesis guide, Prof. Chandan Mazumdar, Department of Computer Science and Engineering, Jadavpur University, Kolkata for his invaluable guidance, constant encouragement and motivating words during the period of my dissertation.

I would also like to thank Dr. Anirban Sengupta, Principal Research Engineer CDCJU, Mr. Subhomoy Karmakar, Research Engineer CDCJU, and Mrs. Puloma Roy, Research Fellow CDCJU, for sharing their knowledge and experience with me and also their immense support and co-operation.

I am thankful to all the teaching and non-teaching staff who helped me to have a smooth journey during the time of my research.

Last but not the least; I would like to thank my family members, classmates, seniors and friends for giving me constant encouragement and mental support throughout my work.

Suchismita Ghosh

University Registration No. : 140747 of 2017-2018

Examination Roll No. : M4CSE19016

Master of Engineering

Department of Computer Science and Engineering

Jadavpur University

Contents

Chapter One: Introduction

1.1. Log: In computing, a log file is a file that records either events that occur in an operating system or other software runs, or messages between different users of a communication software[1]. Log records store the information about the activities done on a machine at a particular time by a user. Each activity is called an event and with each event relevant log record or sequence of log records is generated.

1.2. Logging: The act of keeping a log is called logging. Through logging various types of analysis are done such as auditing, intrusion detection, performance tuning of system. The logging implementation varies with operating system, such as in Unix/Linux platform Syslog daemon (syslogd) is the most common logger while Windows Event Log is the logging implementation in Windows platform. Logging data provides information about the interaction between the production processes along with the timestamp and other necessary information relevant to the event. Therefore, if a company enables data logging within its processes and handles log properly, then it can be traced that what is exactly happening in the production process.

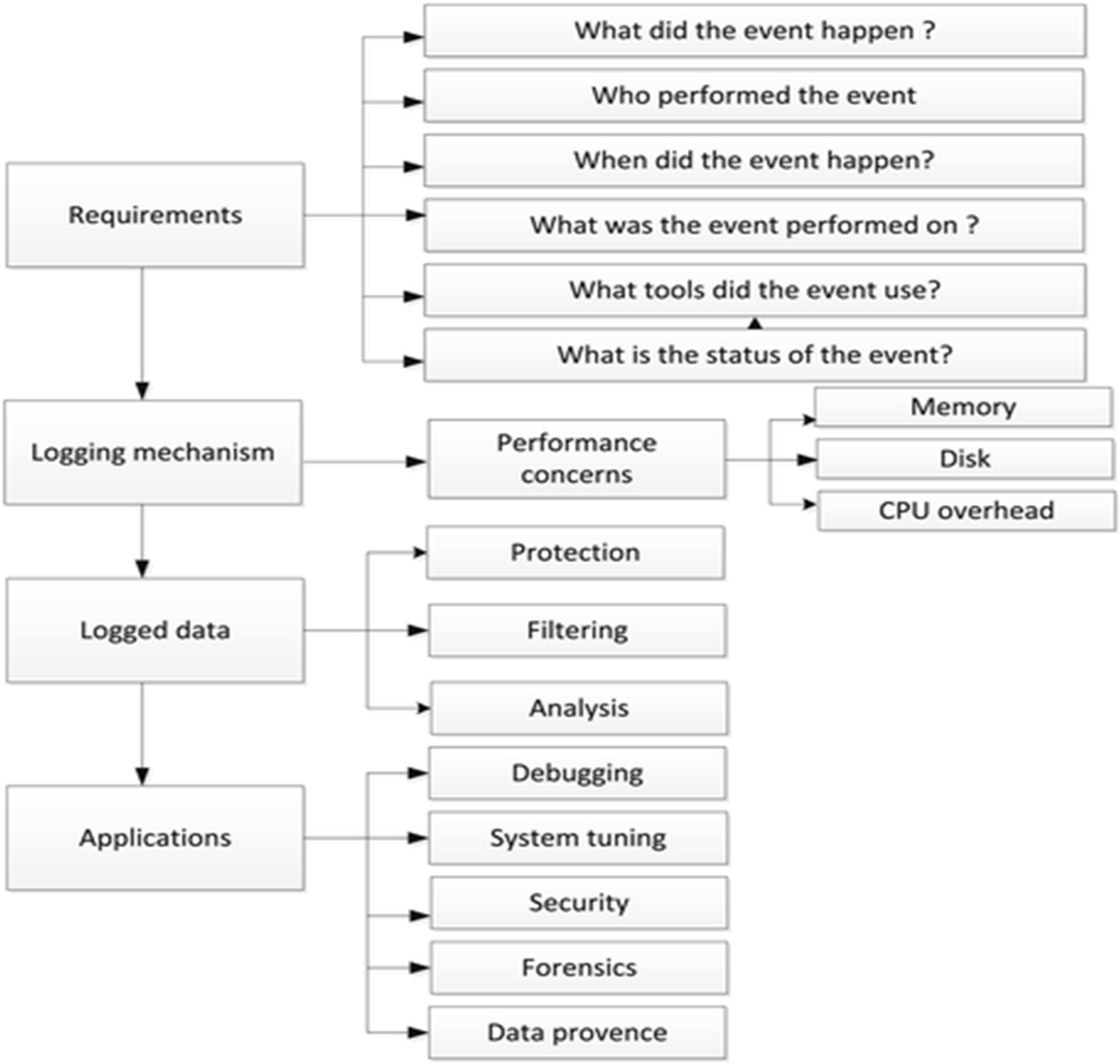
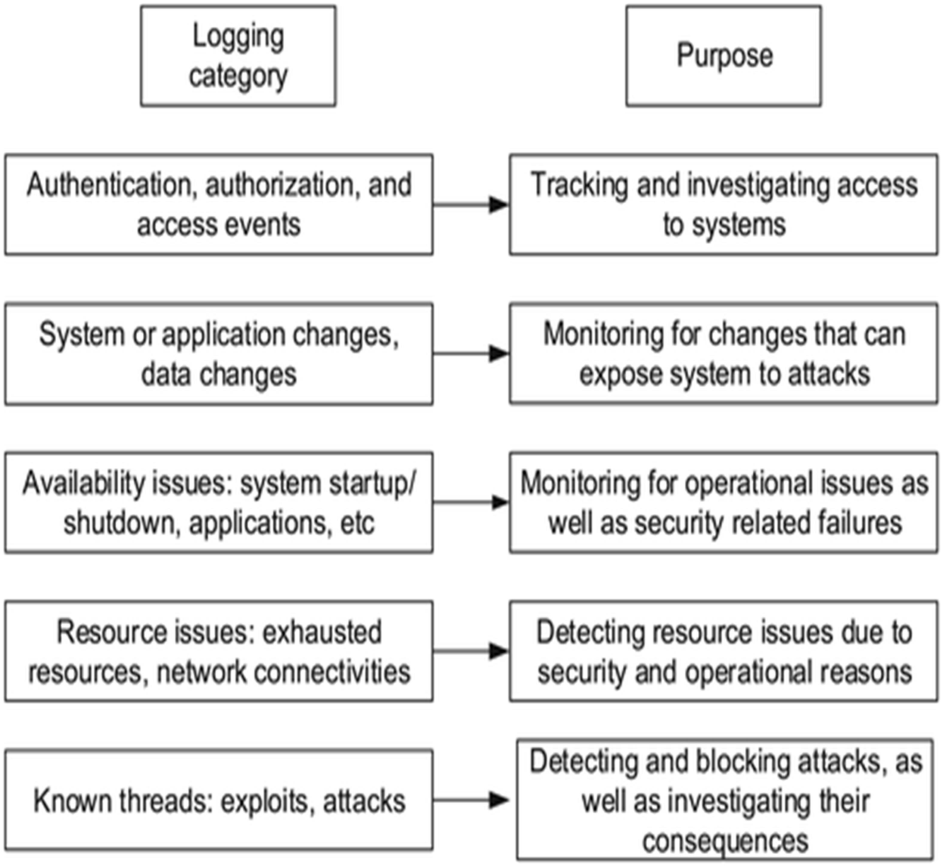


Fig. 1: The Logging Diagram [2]

Using the logged data stored in the system, system problems can be diagnosed and detected as well as the system performance can be optimized. For an example, if a software is installed automatically in the system which may cause low performance of the system, it can be detected from the logged data. Similarly, if there is a batch of activities found in the logged data which is not categorised in the regularly performed activities, then it should be monitored to find out threats for the organization.

Now, it is not required to monitor all the log records generated in the organization. As the volume of the records can be very high for even a single day, it is difficult to store and check all types of logs. To overcome this problem, there are some categories of activities that should be logged.



Logging category	Purpose
Authentication, authorization, and access events	Tracking and investigating access to systems
System or application changes, data changes	Monitoring for changes that can expose system to attacks
Availability issues: system startup/ shutdown, applications, etc	Monitoring for operational issues as well as security related failures
Resource issues: exhausted resources, network connectivities	Detecting resource issues due to security and operational reasons
Known threads: exploits, attacks	Detecting and blocking attacks, as well as investigating their consequences

Fig. 2: Mandate logging categories with corresponding purposes [2]

1.3. <u>Existing tools for log analysis:</u> A log analysis tool collects log, then manages and analyze those logs automatically. Thus it helps to reduce risk by identifying attack attempts, optimize performance of system, track user activity, and helps to meet regulatory compliance. There are a lot of log analysis tools are out there. Some of the log analysis tools are given below:

1.3.1. <u>SolarWinds Event & Log Manager:</u> The key features of this product are[20] :

- Real-time log streaming
- Collection of logs from multiple sources
- Efficient search and filter log data
- Visualization of collected data
- Correlating all logs after centralizing them into a single device
- Forwarding logs to other applications
- Log tagging

This product analyses the log more proactively as it works on real-time data which helps to make an alert before more damage occurs.

1.3.2. Loggly:

Loggly is a cloud based logging management and analytics service provider[21]. It offers log management in a simple way to make it user-friendly. Some of the key features of this products are[22]:

- Collection of any kind of text-based log data dynamically.
- Deployment monitoring
- Versatile alert notification methods (e.g. Email, Slack, GitHub etc.)
- Visualization of data through various types of charts
- Anomaly detection
- Automated parsing different types of log data

1.3.3. ManageEngine EventLog Analyzer:

EventLog Analyzer is a web-based, real-time, log monitoring and compliance management solution for Security Information and Event Management (SIEM) that improves internal network security and helps to comply with the latest IT audit requirement[23]. This product has the following features:

- Collecting all types of machine-generated logs received from systems, network devices and also applications.
- Advanced search and extracting new fields.
- Correlation among events on real-time.
- Monitoring network activities.

- Automated report generation.
- Quick remediation after notifying alerts.

1.3.4. Logstash: It is an open-source data collection and management engine. Logstash provides:

- Real-time pipelining capability
- Monitoring all types of data
- Transform HTTP requests into events
- JDBC interface for better understanding the data
- Various methods for alerting
- Community-extensible and developer-friendly plugin ecosystem[24]

Apart from the mentioned products there are more log analysis and monitoring tools such as Logentries, Graylog, GoAccess, Splunk, LOGalyze, Logmatic, Retrace etc.

The proposed tool includes some distinct features to enhance the power to analyse logs. No prediction is made for developing the algorithm which helps to provide an accurate output.

The rest of the thesis is organised with the description Windows OS and Log structure in chapter 2, introduction of MongoDB in chapter 3, structure of Windows log in chapter 4, some algorithms to analyse logs in chapter 5, the details of implementation of the tool in chapter 6 and the future work that can be applied on this tool in chapter 7.

Chapter Two: Windows OS Structure And Log Generation

2.1. Windows Operating System: A processor in a computer running Windows has two different modes: user mode and kernel mode. The mode switching done by the processor depends on the type of code running on the processor. The major internal components of the Windows operating system are shown in the figure below:

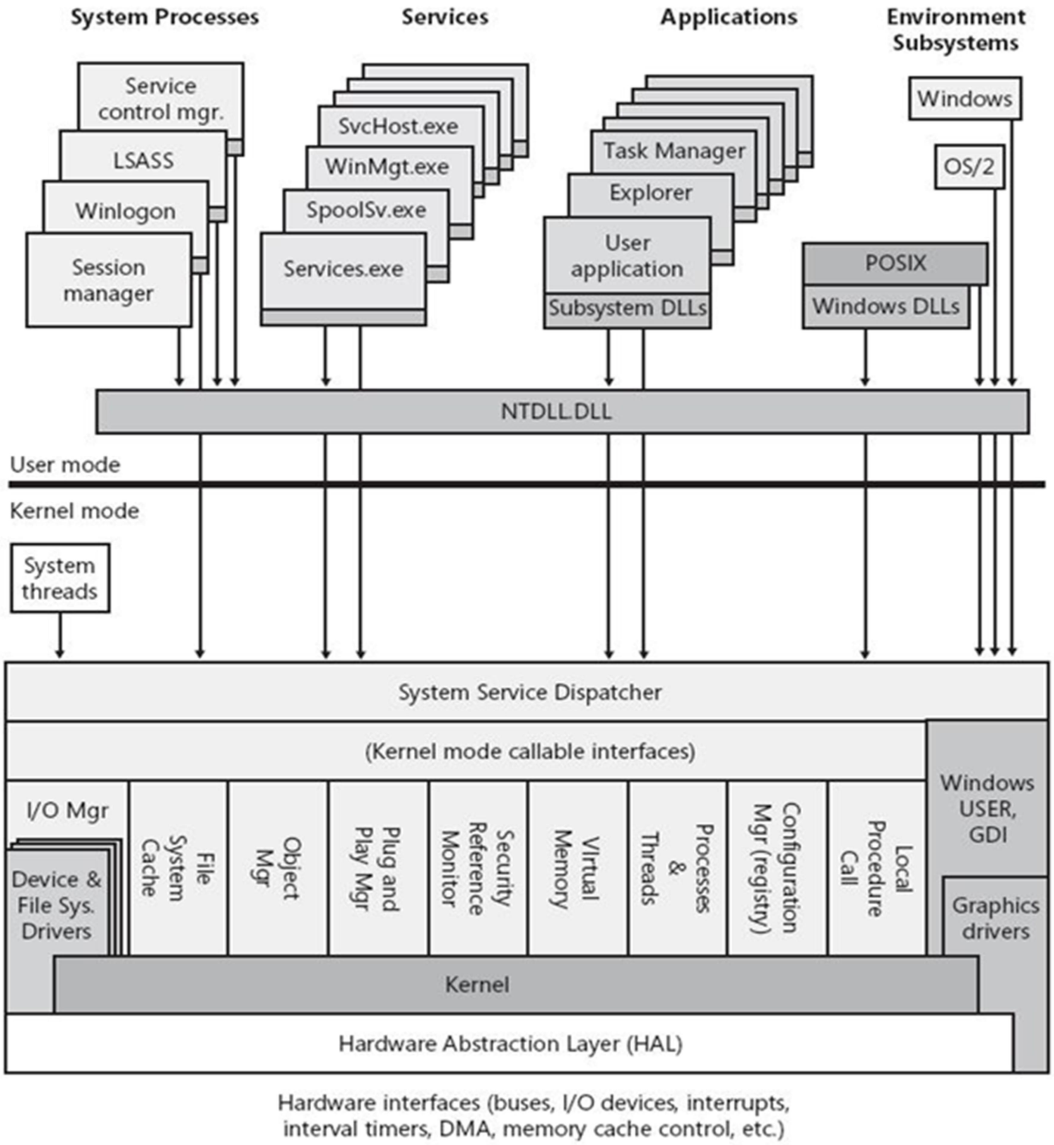


Fig. 3: Components of Windows OS[3]

Applications run in user mode and the core operating system components run in kernel mode. There are two types of drivers depending on their running mode: User-mode drivers and kernel-mode drivers.

2.1.1. User mode: When a user mode application is started, Windows creates a process for the application. Each application will have its own private virtual address space. Because of this, one application can not intervene another while running. As each application runs in isolation, if one is crashed other applications will not be affected.

2.1.2. Kernel mode: Kernel mode uses a single virtual address space for which all components running in kernel mode use a shared virtual address space. If any of the kernel-mode drivers crashes, it will cause damage of the entire operating system.

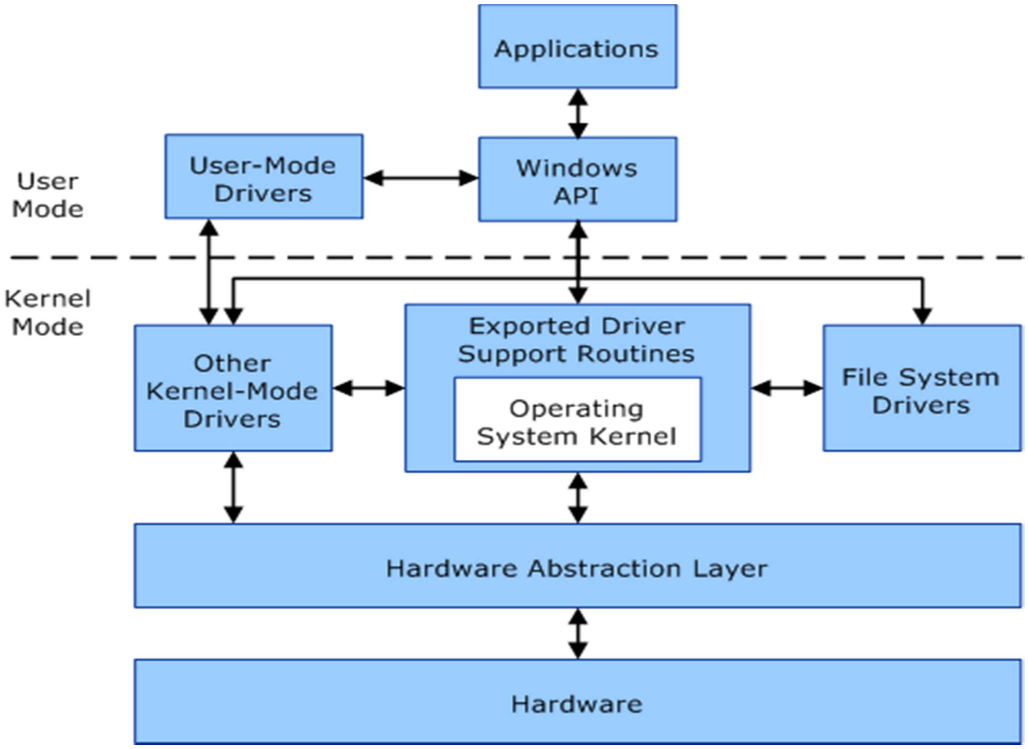


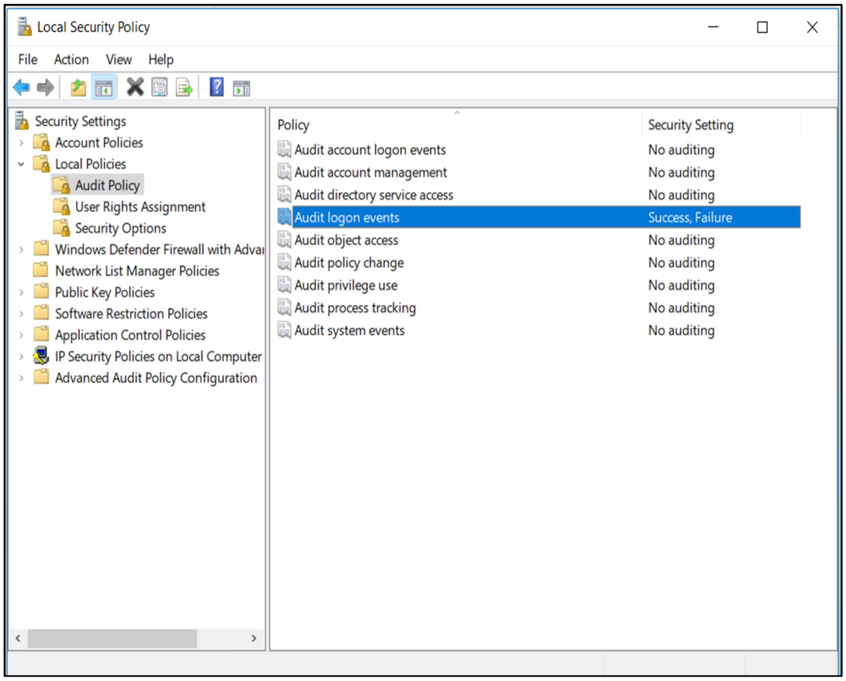
Fig. 4: Communication between user-mode and kernel-mode components[4]

2.2. Windows Log:

The Windows event log is a detailed record of system, security and application notifications stored by the Windows operating system that is used by administrators to diagnose system problems and predict future issues[5]. These event logs are used to record important software as well as hardware actions which helps the administrator to troubleshoot issues with the operating system.

C2-level security requirements specify that system administrators must be able to audit security-related events and that access to this audit data must be limited to authorized administrators. The Windows API provides functions enabling an administrator to monitor security-related events. The security descriptor for a securable object can have a system access control list (SACL). A SACL contains access control entries (ACEs) that specify the types of access attempts that generate audit reports[6].

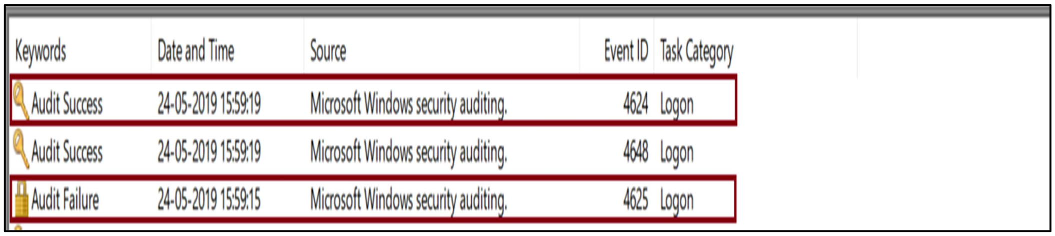
To generate log, audit policies need to be configured manually. These audit policies can be found by following this path: Administrative Tools -> Local Policies -> Audit Policy. After a specific policy is configured for an attempt, whenever an event related to that attempt is occurred, a log record is generated. Generally there are two types of attempts: success and failure. A screenshot of configuring audit logon events policy is given below:


Local Security Policy
File
Action
View
Help
Security Settings
Account Policies
Local Policies
Audit Policy
User Rights Assignment
Security Options
Windows Defender Firewall with Advai
Network List Manager Policies
Public Key Policies
Software Restriction Policies
Application Control Policies
IP Security Policies on Local Computer
Advanced Audit Policy Configuration

Policy	Security Setting
Audit account logon events	No auditing
Audit account management	No auditing
Audit directory service access	No auditing
Audit logon events	Success, Failure
Audit object access	No auditing
Audit policy change	No auditing
Audit privilege use	No auditing
Audit process tracking	No auditing
Audit system events	No auditing

Fig. 5: Audit policy configuration

After configuring the policy, whenever the system is logged on successfully or user attempts to log on the system with wrong password, a corresponding log will be generated. Similarly, when the user account's logon session is terminated, a logoff record will also be generated in the log file.

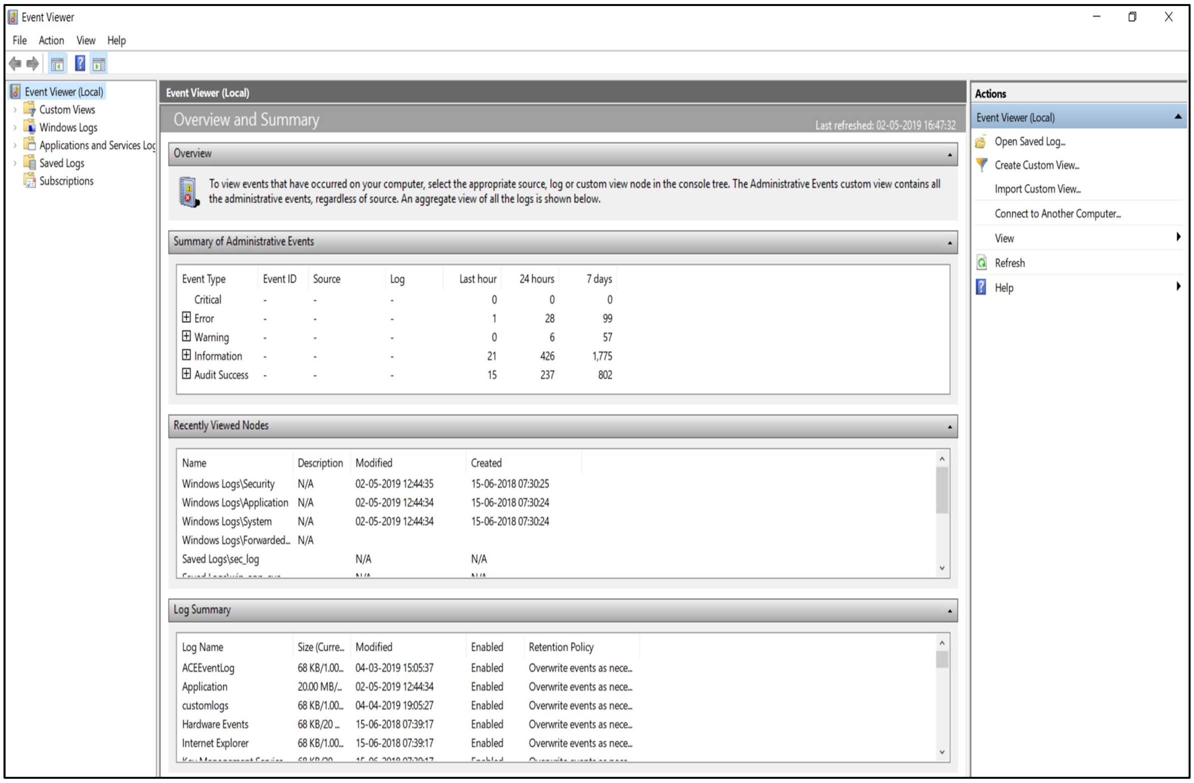


Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	24-05-2019 15:59:19	Microsoft Windows security auditing.	4624	Logon
Audit Success	24-05-2019 15:59:19	Microsoft Windows security auditing.	4648	Logon
Audit Failure	24-05-2019 15:59:15	Microsoft Windows security auditing.	4625	Logon

Fig. 6: Generated log after configuring audit policy

As it can be seen that both the audit success and failure records were generated during logon.

2.2.1. <u>Collection of Windows Logs:</u> **Event Viewer** is a component of Microsoft's Windows NT line of operating systems that lets administrators and users view the event logs on a local or remote machine[7].



Last refreshed: 02-05-2019 16:47:32

To view events that have occurred on your computer, select the appropriate source, log or custom view node in the console tree. The Administrative Events custom view contains all the administrative events, regardless of source. An aggregate view of all the logs is shown below.

Event Type	Event ID	Source	Log	Last hour	24 hours	7 days
Critical	-	-	-	0	0	0
⊞ Error	-	-	-	1	28	99
⊞ Warning	-	-	-	0	6	57
⊞ Information	-	-	-	21	426	1,775
⊞ Audit Success	-	-	-	15	237	802

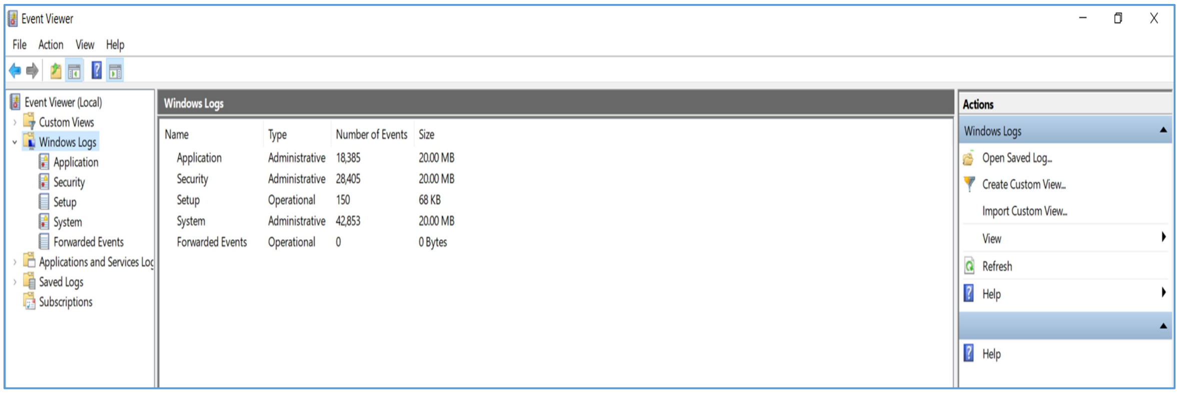
Name	Description	Modified	Created
Windows Logs\Security	N/A	02-05-2019 12:44:35	15-06-2018 07:30:25
Windows Logs\Application	N/A	02-05-2019 12:44:34	15-06-2018 07:30:24
Windows Logs\System	N/A	02-05-2019 12:44:34	15-06-2018 07:30:24
Windows Logs\Forwarded...	N/A		
Saved Logs\sec_log		N/A	N/A

Log Name	Size (Curre...	Modified	Enabled	Retention Policy
ACEEventLog	68 KB/1.00...	04-03-2019 15:05:37	Enabled	Overwrite events as nece...
Application	20.00 MB/...	02-05-2019 12:44:34	Enabled	Overwrite events as nece...
customlogs	68 KB/1.00...	04-04-2019 19:05:27	Enabled	Overwrite events as nece...
Hardware Events	68 KB/20 ...	15-06-2018 07:39:17	Enabled	Overwrite events as nece...
Internet Explorer	68 KB/1.00...	15-06-2018 07:39:17	Enabled	Overwrite events as nece...

Fig. 7: The Event Viewer window

The ‘Windows Logs’ folder is subdivided into five types of logs as given below:

Application, Security, Setup, System, Forwarded Events.

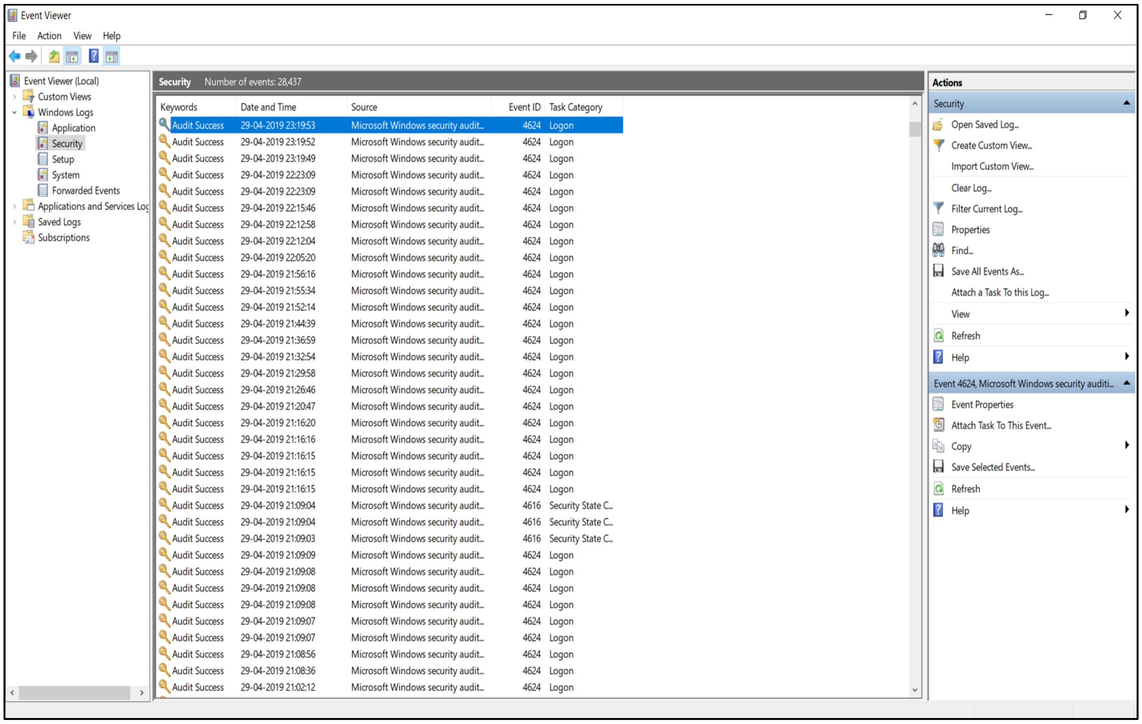


Name	Type	Number of Events	Size
Application	Administrative	18,385	20.00 MB
Security	Administrative	28,405	20.00 MB
Setup	Operational	150	68 KB
System	Administrative	42,853	20.00 MB
Forwarded Events	Operational	0	0 Bytes

Fig. 8: Divisions of Windows Logs

After opening any of these five files, it can be seen that each event record is stored in five different fields namely –

- Keywords
- Date and Time
- Source
- Event ID
- Task Category



Security Number of events: 28,437

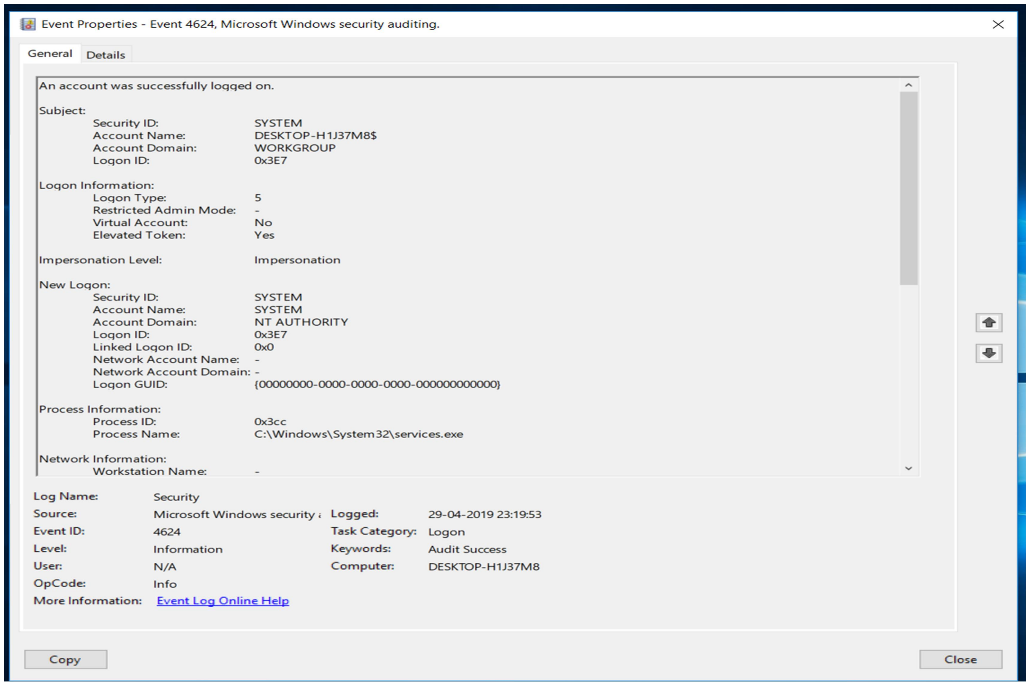
Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	29-04-2019 23:19:53	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 23:19:52	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 23:19:49	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 22:23:09	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 22:23:09	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 22:15:46	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 22:12:58	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 22:12:04	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 22:05:20	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:56:16	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:55:34	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:52:14	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:44:39	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:36:59	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:32:54	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:29:58	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:26:46	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:20:47	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:16:20	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:16:16	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:16:15	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:16:15	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:16:15	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:04	Microsoft Windows security audit...	4616	Security State C...
Audit Success	29-04-2019 21:09:04	Microsoft Windows security audit...	4616	Security State C...
Audit Success	29-04-2019 21:09:03	Microsoft Windows security audit...	4616	Security State C...
Audit Success	29-04-2019 21:09:09	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:08	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:08	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:08	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:07	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:07	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:08:56	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:08:36	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:02:12	Microsoft Windows security audit...	4624	Logon

Fig. 9: Security Logs

After clicking on the ‘Event Properties’ option for a particular event another window is opened with two tabs- General and Details as shown below.

The General tab is divided into two portions:

The above portion is the ‘Message’ part of the event and the below portion provides the values of other relevant fields.


Event Properties - Event 4624, Microsoft Windows security auditing.
General
Details
Copy
Close

```
An account was successfully logged on.

Subject:
        Security ID:                SYSTEM
        Account Name:               DESKTOP-H1J37M8$
        Account Domain:             WORKGROUP
        Logon ID:                   0x3E7

Logon Information:
        Logon Type:                 5
        Restricted Admin Mode:      -
        Virtual Account:            No
        Elevated Token:             Yes

Impersonation Level:                Impersonation

New Logon:
        Security ID:                SYSTEM
        Account Name:               SYSTEM
        Account Domain:             NT AUTHORITY
        Logon ID:                   0x3E7
        Linked Logon ID:            0x0
        Network Account Name:       -
        Network Account Domain:     -
        Logon GUID:                 {00000000-0000-0000-0000-000000000000}

Process Information:
        Process ID:                 0x3cc
        Process Name:               C:\Windows\System32\services.exe

Network Information:
        Workstation Name:           -
```

```
Log Name:          Security
Source:            Microsoft Windows security a   Logged:          29-04-2019 23:19:53
Event ID:          4624                           Task Category:   Logon
Level:             Information                    Keywords:        Audit Success
User:              N/A                            Computer:        DESKTOP-H1J37M8
OpCode:            Info
More Information:  Event Log Online Help
```

Fig. 9: A screenshot of General tab

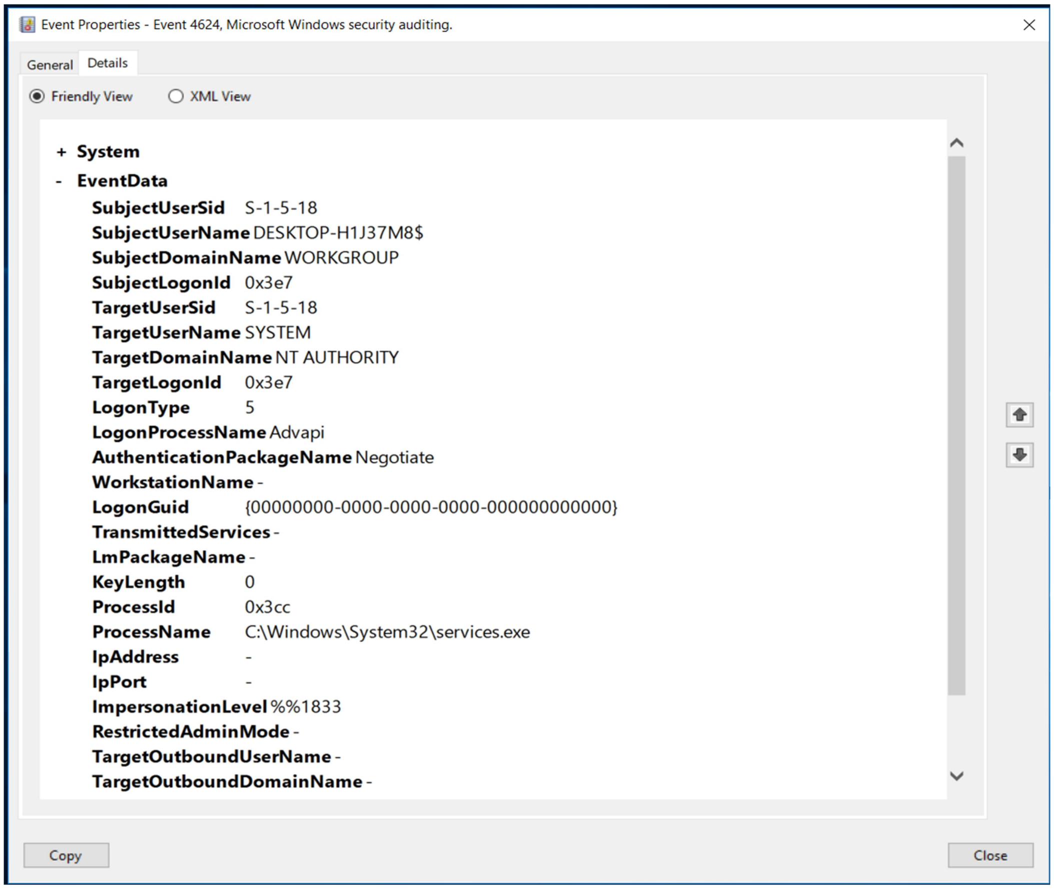
In the 'Details' tab the log data for an event is divided into two parts:

- System, and
- EventData

Two types of views for the data are provided:

- Friendly View
- XML View

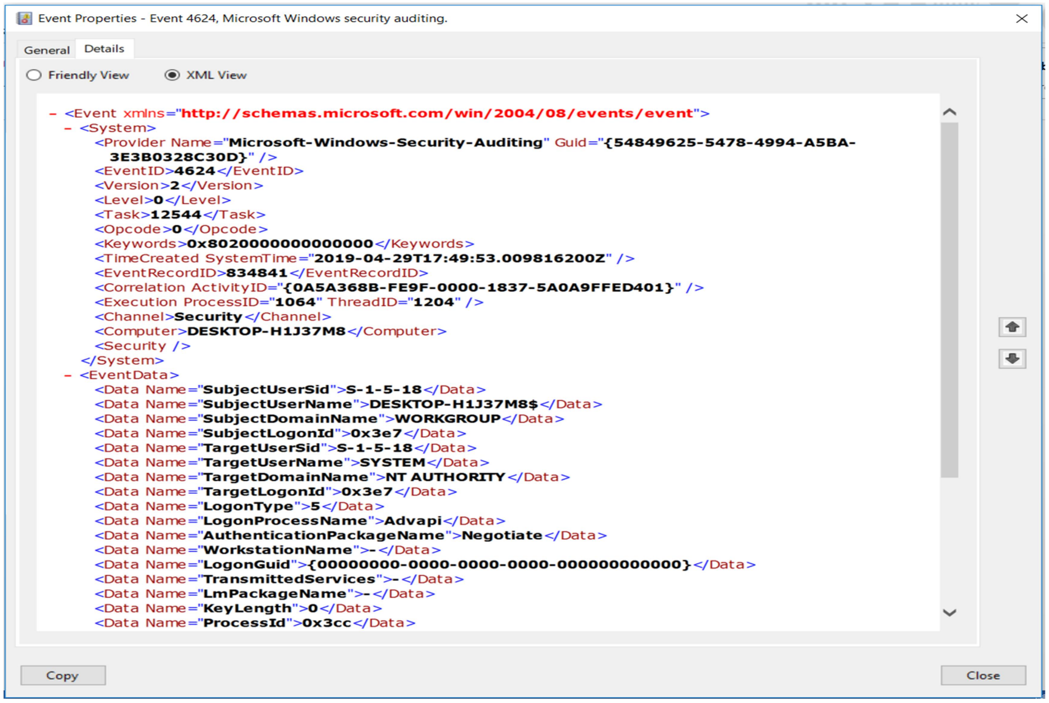
Both of these types are shown below.


Event Properties - Event 4624, Microsoft Windows security auditing.
General
Details
Copy
Close

● Friendly View ○ XML View

```
+ System
- EventData
    SubjectUserSid        S-1-5-18
    SubjectUserName       DESKTOP-H1J37M8$
    SubjectDomainName     WORKGROUP
    SubjectLogonId        0x3e7
    TargetUserSid         S-1-5-18
    TargetUserName        SYSTEM
    TargetDomainName      NT AUTHORITY
    TargetLogonId         0x3e7
    LogonType             5
    LogonProcessName      Advapi
    AuthenticationPackageName Negotiate
    WorkstationName       -
    LogonGuid             {00000000-0000-0000-0000-000000000000}
    TransmittedServices   -
    LmPackageName         -
    KeyLength             0
    ProcessId             0x3cc
    ProcessName           C:\Windows\System32\services.exe
    IpAddress             -
    IpPort                -
    ImpersonationLevel    %%1833
    RestrictedAdminMode   -
    TargetOutboundUserName -
    TargetOutboundDomainName -
```

Fig. 10: Friendly view of event log


Event Properties - Event 4624, Microsoft Windows security auditing.
General
Details
Copy
Close

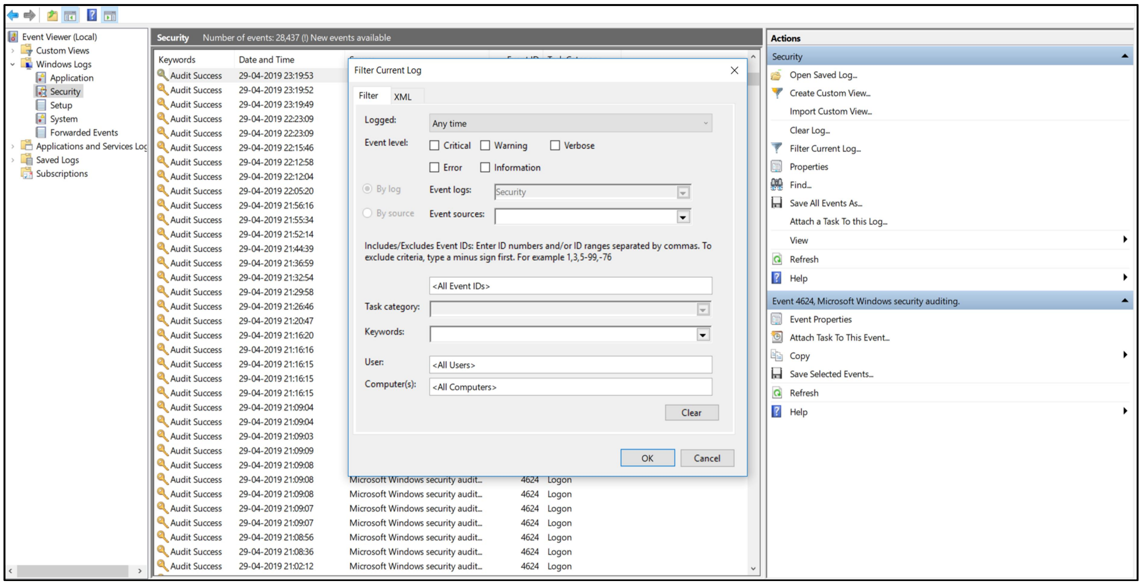
○ Friendly View ● XML View

```
- <Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
  - <System>
      <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-
        3E3B0328C30D}" />
      <EventID>4624</EventID>
      <Version>2</Version>
      <Level>0</Level>
      <Task>12544</Task>
      <Opcode>0</Opcode>
      <Keywords>0x8020000000000000</Keywords>
      <TimeCreated SystemTime="2019-04-29T17:49:53.009816200Z" />
      <EventRecordID>834841</EventRecordID>
      <Correlation ActivityID="{0A5A368B-FE9F-0000-1837-5A0A9FFED401}" />
      <Execution ProcessID="1064" ThreadID="1204" />
      <Channel>Security</Channel>
      <Computer>DESKTOP-H1J37M8</Computer>
      <Security />
    </System>
  - <EventData>
      <Data Name="SubjectUserSid">S-1-5-18</Data>
      <Data Name="SubjectUserName">DESKTOP-H1J37M8$</Data>
      <Data Name="SubjectDomainName">WORKGROUP</Data>
      <Data Name="SubjectLogonId">0x3e7</Data>
      <Data Name="TargetUserSid">S-1-5-18</Data>
      <Data Name="TargetUserName">SYSTEM</Data>
      <Data Name="TargetDomainName">NT AUTHORITY</Data>
      <Data Name="TargetLogonId">0x3e7</Data>
      <Data Name="LogonType">5</Data>
      <Data Name="LogonProcessName">Advapi</Data>
      <Data Name="AuthenticationPackageName">Negotiate</Data>
      <Data Name="WorkstationName">-</Data>
      <Data Name="LogonGuid">{00000000-0000-0000-0000-000000000000}</Data>
      <Data Name="TransmittedServices">-</Data>
      <Data Name="LmPackageName">-</Data>
      <Data Name="KeyLength">0</Data>
      <Data Name="ProcessId">0x3cc</Data>
```

Fig. 11: XML view of event log

The Event Viewer also provides other features like

- Advanced filtering the current log: Events can easily be filtered by any criteria including event description text. It can be done either by applying the provided filters or by giving XPath expression in the XML Query manually. The filters are reusable – they can be saved as a file and applied to other event logs.



Security Number of events: 28,437 (!) New events available

Keywords	Date and Time			
Audit Success	29-04-2019 23:19:53			
Audit Success	29-04-2019 23:19:52			
Audit Success	29-04-2019 23:19:49			
Audit Success	29-04-2019 22:23:09			
Audit Success	29-04-2019 22:23:09			
Audit Success	29-04-2019 22:15:46			
Audit Success	29-04-2019 22:12:58			
Audit Success	29-04-2019 22:12:04			
Audit Success	29-04-2019 22:05:20			
Audit Success	29-04-2019 21:56:16			
Audit Success	29-04-2019 21:55:34			
Audit Success	29-04-2019 21:52:14			
Audit Success	29-04-2019 21:44:39			
Audit Success	29-04-2019 21:36:59			
Audit Success	29-04-2019 21:32:54			
Audit Success	29-04-2019 21:29:58			
Audit Success	29-04-2019 21:26:46			
Audit Success	29-04-2019 21:20:47			
Audit Success	29-04-2019 21:16:20			
Audit Success	29-04-2019 21:16:16			
Audit Success	29-04-2019 21:16:15			
Audit Success	29-04-2019 21:16:15			
Audit Success	29-04-2019 21:16:15			
Audit Success	29-04-2019 21:09:04			
Audit Success	29-04-2019 21:09:04			
Audit Success	29-04-2019 21:09:03			
Audit Success	29-04-2019 21:09:09			
Audit Success	29-04-2019 21:09:08			
Audit Success	29-04-2019 21:09:08	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:08	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:07	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:07	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:08:56	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:08:36	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:02:12	Microsoft Windows security audit...	4624	Logon

Filter Current Log

Filter | XML

Logged: Any time

Event level: □ Critical □ Warning □ Verbose □ Error □ Information

● By log Event logs: Security

○ By source Event sources:

Includes/Excludes Event IDs: Enter ID numbers and/or ID ranges separated by commas. To exclude criteria, type a minus sign first. For example 1,3,5-99,-76

<All Event IDs>

Task category:

Keywords:

User: <All Users>

Computer(s): <All Computers>

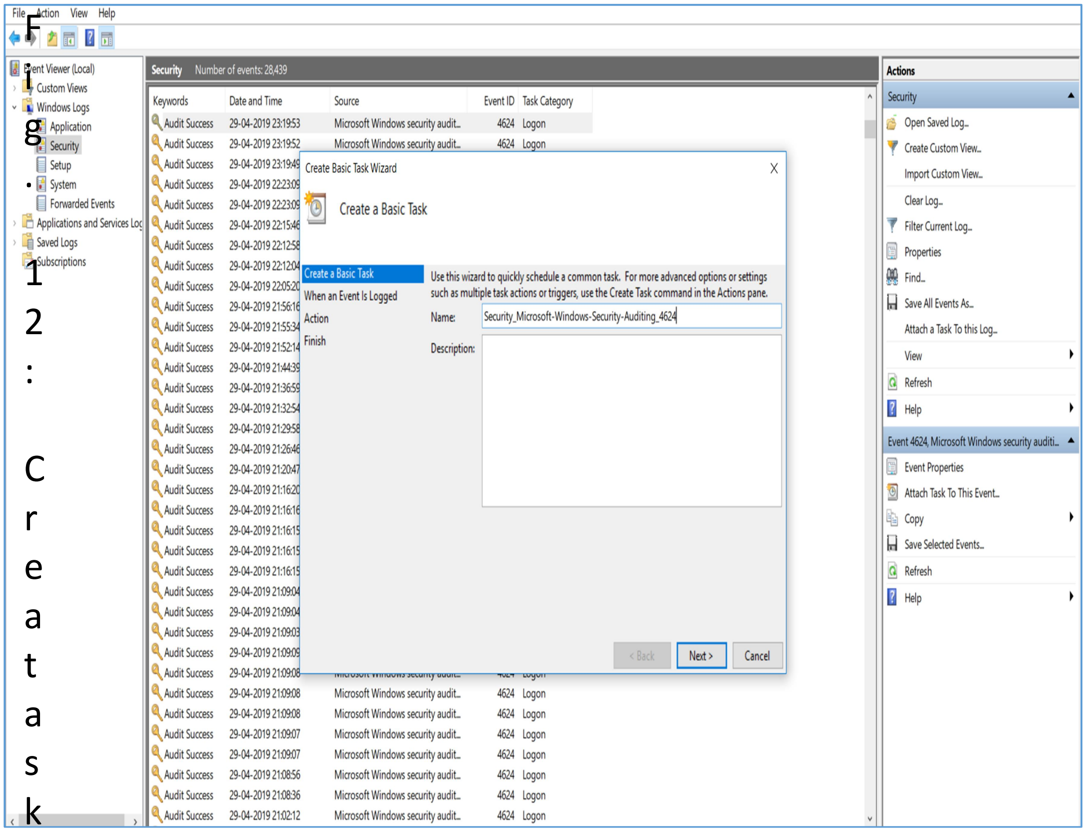
Clear

OK Cancel

Fig. 11: Filtration of log

- Run a task in response to an event: Event viewer lets the user take actions in response to an event through the option 'Attach Task To This Event'. There are three kinds of actions can be taken –
    I. Start a program
    II. Send an e-mail (deprecated)
    III. Display a message (deprecated)

With the help of this feature whenever the expected event occurs, the mentioned action will be triggered. This helps to automate some tasks using built-in scheduler.



F
i
g
1
2
:
C
r
e
a
t
a
s
k

Security Number of events: 28,439

Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	29-04-2019 23:19:53	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 23:19:52	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 23:19:49			
Audit Success	29-04-2019 22:23:09			
Audit Success	29-04-2019 22:23:09			
Audit Success	29-04-2019 22:15:46			
Audit Success	29-04-2019 22:12:58			
Audit Success	29-04-2019 22:12:04			
Audit Success	29-04-2019 22:05:20			
Audit Success	29-04-2019 21:56:16			
Audit Success	29-04-2019 21:55:34			
Audit Success	29-04-2019 21:52:14			
Audit Success	29-04-2019 21:44:39			
Audit Success	29-04-2019 21:36:59			
Audit Success	29-04-2019 21:32:54			
Audit Success	29-04-2019 21:29:58			
Audit Success	29-04-2019 21:26:46			
Audit Success	29-04-2019 21:20:47			
Audit Success	29-04-2019 21:16:20			
Audit Success	29-04-2019 21:16:16			
Audit Success	29-04-2019 21:16:15			
Audit Success	29-04-2019 21:16:15			
Audit Success	29-04-2019 21:16:15			
Audit Success	29-04-2019 21:09:04			
Audit Success	29-04-2019 21:09:04			
Audit Success	29-04-2019 21:09:03			
Audit Success	29-04-2019 21:09:09			
Audit Success	29-04-2019 21:09:08			
Audit Success	29-04-2019 21:09:08	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:08	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:07	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:09:07	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:08:56	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:08:36	Microsoft Windows security audit...	4624	Logon
Audit Success	29-04-2019 21:02:12	Microsoft Windows security audit...	4624	Logon

Create Basic Task Wizard

Create a Basic Task

Create a Basic Task
When an Event Is Logged
Action
Finish

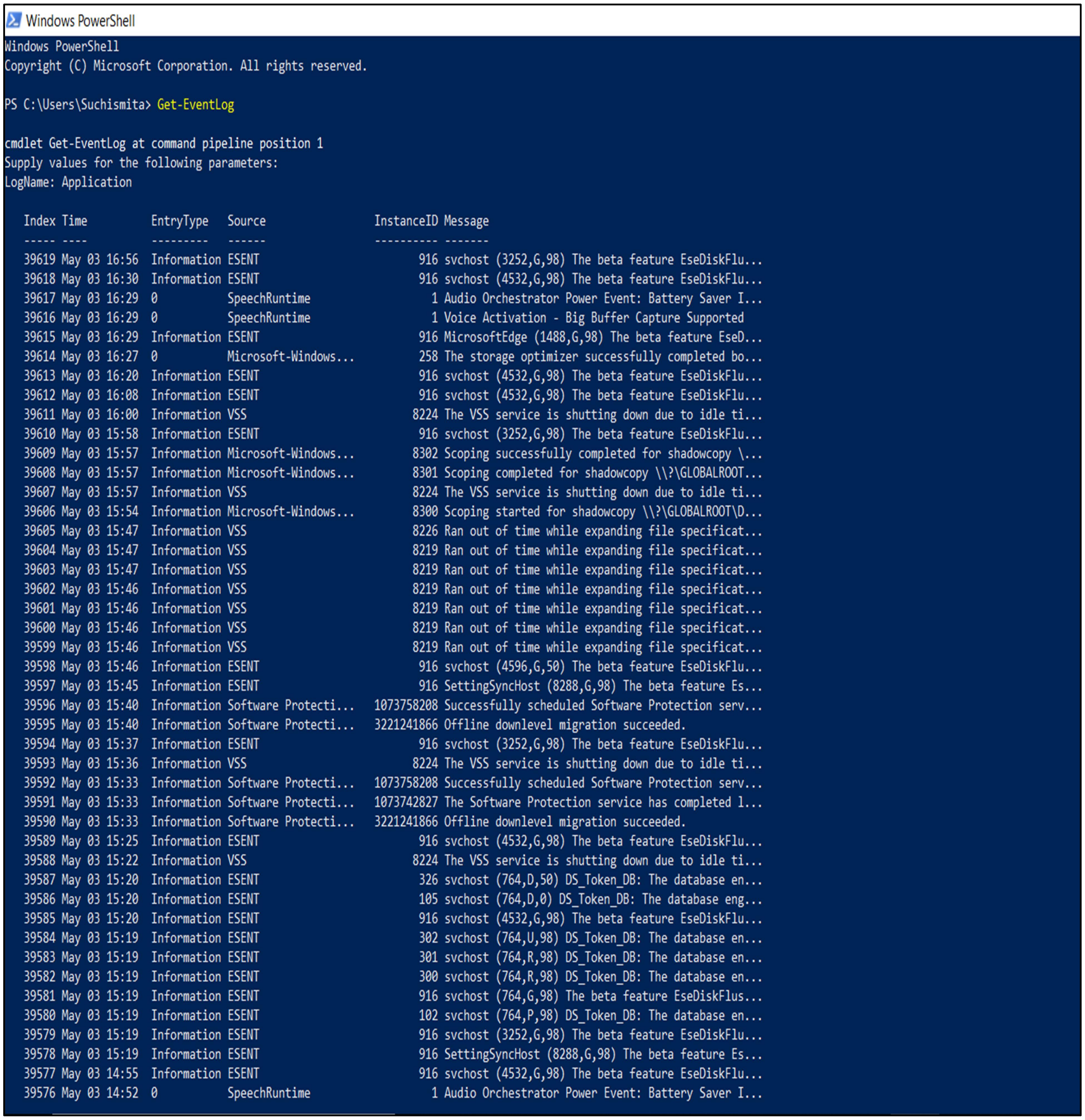
Use this wizard to quickly schedule a common task. For more advanced options or settings such as multiple task actions or triggers, use the Create Task command in the Actions pane.

Name: Security_Microsoft-Windows-Security-Auditing_4624

Description:

Fig. 12: Creating a task

However, the Event Viewer is not the only way to get Windows event logs. The logged events can also be accessed through Powershell. The **Get-EventLog cmdlet** gets events and event logs from local and remote computers. By default, Get-EventLog gets logs from the local computer. To get logs from remote computers, the **ComputerName** parameter is used. To get the log of a specific type (Application or System or Security), the LogName parameter value has to be provided. Get-EventLog also lets the user filter the events by matching the parameters with the specified property values.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Suchismita> Get-EventLog

cmdlet Get-EventLog at command pipeline position 1
Supply values for the following parameters:
LogName: Application

   Index Time          EntryType   Source                 InstanceID Message
   ----- ----          ---------   ------                 ---------- -------
   39619 May 03 16:56  Information ESENT                         916 svchost (3252,G,98) The beta feature EseDiskFlu...
   39618 May 03 16:30  Information ESENT                         916 svchost (4532,G,98) The beta feature EseDiskFlu...
   39617 May 03 16:29  0           SpeechRuntime                   1 Audio Orchestrator Power Event: Battery Saver I...
   39616 May 03 16:29  0           SpeechRuntime                   1 Voice Activation - Big Buffer Capture Supported
   39615 May 03 16:29  Information ESENT                         916 MicrosoftEdge (1488,G,98) The beta feature EseD...
   39614 May 03 16:27  0           Microsoft-Windows...          258 The storage optimizer successfully completed bo...
   39613 May 03 16:20  Information ESENT                         916 svchost (4532,G,98) The beta feature EseDiskFlu...
   39612 May 03 16:08  Information ESENT                         916 svchost (4532,G,98) The beta feature EseDiskFlu...
   39611 May 03 16:00  Information VSS                          8224 The VSS service is shutting down due to idle ti...
   39610 May 03 15:58  Information ESENT                         916 svchost (3252,G,98) The beta feature EseDiskFlu...
   39609 May 03 15:57  Information Microsoft-Windows...         8302 Scoping successfully completed for shadowcopy \...
   39608 May 03 15:57  Information Microsoft-Windows...         8301 Scoping completed for shadowcopy \\?\GLOBALROOT...
   39607 May 03 15:57  Information VSS                          8224 The VSS service is shutting down due to idle ti...
   39606 May 03 15:54  Information Microsoft-Windows...         8300 Scoping started for shadowcopy \\?\GLOBALROOT\D...
   39605 May 03 15:47  Information VSS                          8226 Ran out of time while expanding file specificat...
   39604 May 03 15:47  Information VSS                          8219 Ran out of time while expanding file specificat...
   39603 May 03 15:47  Information VSS                          8219 Ran out of time while expanding file specificat...
   39602 May 03 15:46  Information VSS                          8219 Ran out of time while expanding file specificat...
   39601 May 03 15:46  Information VSS                          8219 Ran out of time while expanding file specificat...
   39600 May 03 15:46  Information VSS                          8219 Ran out of time while expanding file specificat...
   39599 May 03 15:46  Information VSS                          8219 Ran out of time while expanding file specificat...
   39598 May 03 15:46  Information ESENT                         916 svchost (4596,G,50) The beta feature EseDiskFlu...
   39597 May 03 15:45  Information ESENT                         916 SettingSyncHost (8288,G,98) The beta feature Es...
   39596 May 03 15:40  Information Software Protecti...   1073758208 Successfully scheduled Software Protection serv...
   39595 May 03 15:40  Information Software Protecti...   3221241866 Offline downlevel migration succeeded.
   39594 May 03 15:37  Information ESENT                         916 svchost (3252,G,98) The beta feature EseDiskFlu...
   39593 May 03 15:36  Information VSS                          8224 The VSS service is shutting down due to idle ti...
   39592 May 03 15:33  Information Software Protecti...   1073758208 Successfully scheduled Software Protection serv...
   39591 May 03 15:33  Information Software Protecti...   1073742827 The Software Protection service has completed l...
   39590 May 03 15:33  Information Software Protecti...   3221241866 Offline downlevel migration succeeded.
   39589 May 03 15:25  Information ESENT                         916 svchost (4532,G,98) The beta feature EseDiskFlu...
   39588 May 03 15:22  Information VSS                          8224 The VSS service is shutting down due to idle ti...
   39587 May 03 15:20  Information ESENT                         326 svchost (764,D,50) DS_Token_DB: The database en...
   39586 May 03 15:20  Information ESENT                         105 svchost (764,D,0) DS_Token_DB: The database eng...
   39585 May 03 15:20  Information ESENT                         916 svchost (4532,G,98) The beta feature EseDiskFlu...
   39584 May 03 15:19  Information ESENT                         302 svchost (764,U,98) DS_Token_DB: The database en...
   39583 May 03 15:19  Information ESENT                         301 svchost (764,R,98) DS_Token_DB: The database en...
   39582 May 03 15:19  Information ESENT                         300 svchost (764,R,98) DS_Token_DB: The database en...
   39581 May 03 15:19  Information ESENT                         916 svchost (764,G,98) The beta feature EseDiskFlus...
   39580 May 03 15:19  Information ESENT                         102 svchost (764,P,98) DS_Token_DB: The database en...
   39579 May 03 15:19  Information ESENT                         916 svchost (3252,G,98) The beta feature EseDiskFlu...
   39578 May 03 15:19  Information ESENT                         916 SettingSyncHost (8288,G,98) The beta feature Es...
   39577 May 03 14:55  Information ESENT                         916 svchost (4532,G,98) The beta feature EseDiskFlu...
   39576 May 03 14:52  0           SpeechRuntime                   1 Audio Orchestrator Power Event: Battery Saver I...
```

Fig. 13: Event logs shown in Powershell

Chapter Three: Introduction of MongoDB

3.1. MongoDB is a cross-platform document-oriented NoSQL database program that uses JSON-like documents with schemata[8]. MongoDB documents are composed of key-value pairs like the following structure:

```
{
  key1: value1,
  key2: value2,
  key3: value3,
  ...
  keyN: valueN
}
```

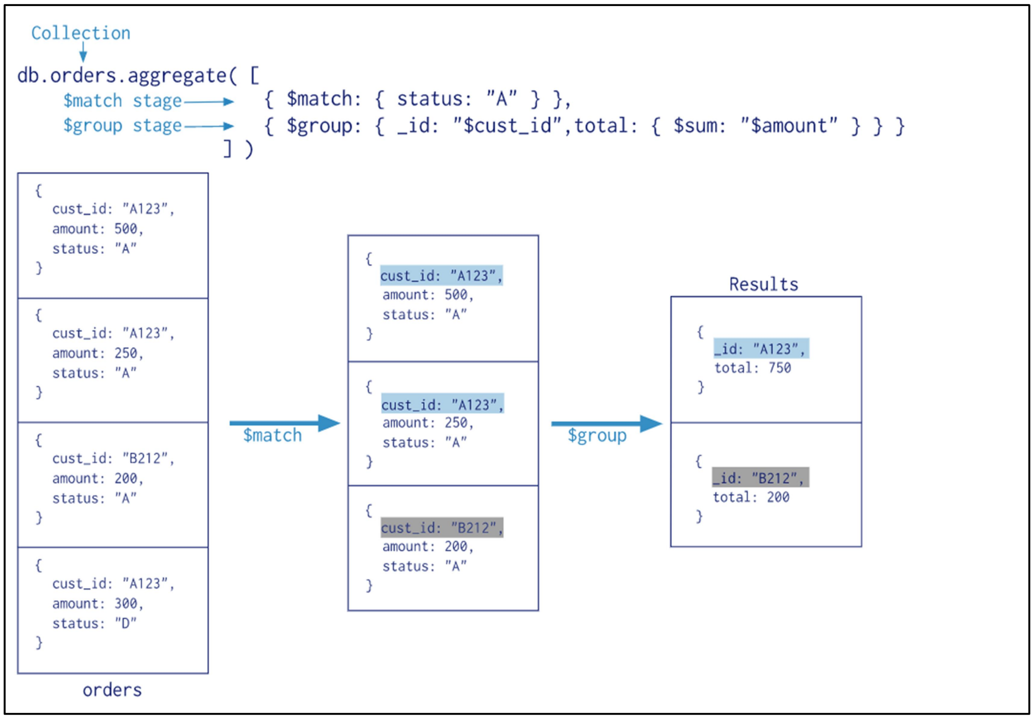
While inserting the data, if any primary key is not mentioned for the database, the '_id' field is reserved for the primary key. If any unique value is not provided for the '_id' field, it stores the "MongoDB ObjectId", generated by MongoDB driver, by default. Though the keys may vary from document to document, the '_id' is always the first field in every document. There are three editions of MongoDB [8] :

- MongoDB Community Server
- MongoDB Enterprise Server
- MongoDB Atlas

3.1.1. Features: The main features of MongoDB are

- Schema-less database: MongoDB is a schema-less database written in C++. MongoDB stores data in BSON format in a flexible way that accepts unstructured data. That means, before inserting the data the table’s schema does not need to be determined and declared.

- Ad-hoc queries: MongoDB supports field, range query, and regular expression searches. Queries can return specific fields of documents and also include user-defined JavaScript functions. Queries can also be configured to return a random sample of results of a given size[8].
- Aggregation[9]: MongoDB’s aggregation framework is modelled on the concept of data processing pipelines. Documents enter a multi-stage pipeline that transforms the documents into an aggregated result.

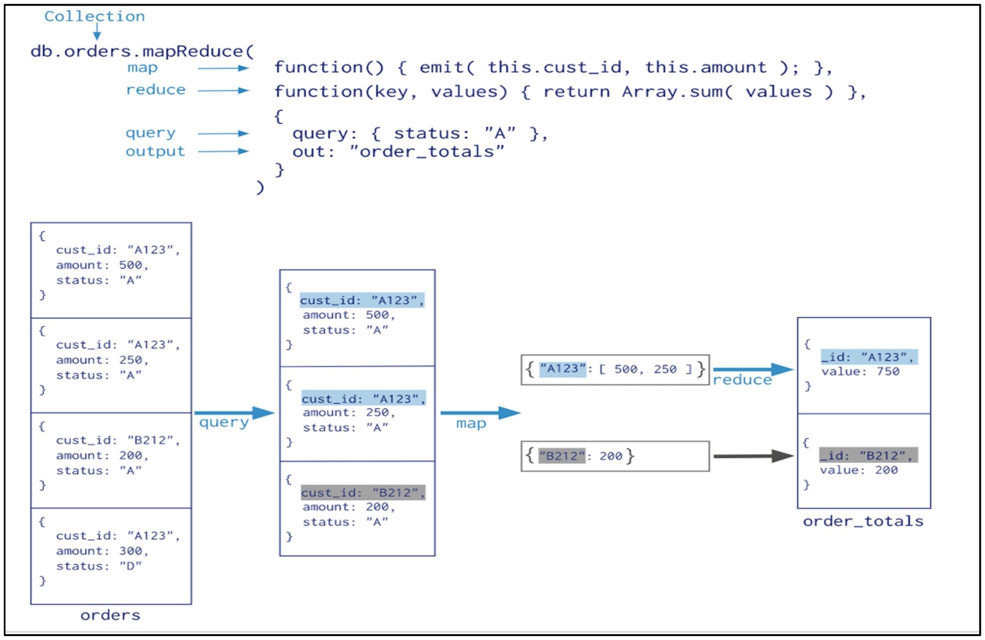


```
db.orders.aggregate( [
    { $match: { status: "A" } },
    { $group: { _id: "$cust_id",total: { $sum: "$amount" } } }
] )
```

Collection
$match stage
$group stage
{ cust_id: "A123", amount: 500, status: "A" }
{ cust_id: "A123", amount: 250, status: "A" }
{ cust_id: "B212", amount: 200, status: "A" }
{ cust_id: "A123", amount: 300, status: "D" }
orders
$match
{ cust_id: "A123", amount: 500, status: "A" }
{ cust_id: "A123", amount: 250, status: "A" }
{ cust_id: "B212", amount: 200, status: "A" }
$group
Results
{ _id: "A123", total: 750 }
{ _id: "B212", total: 200 }

Fig. 14: Aggregation using Mongo query

Map-reduce operations are also provided by MongoDB.

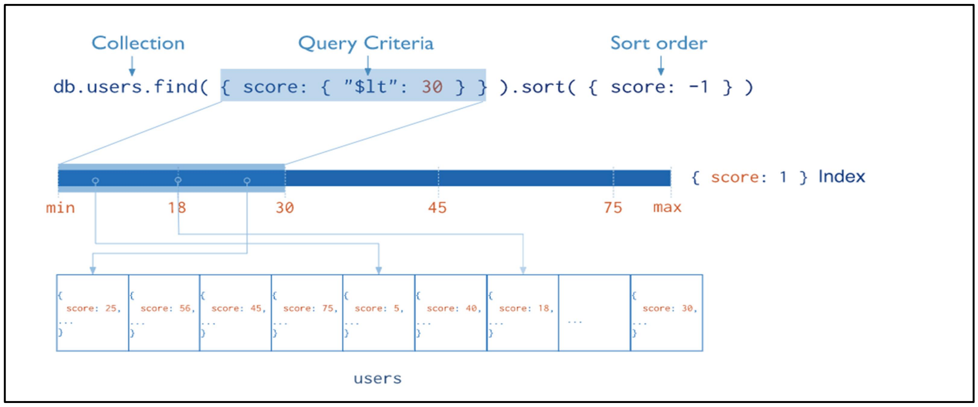


```
db.orders.mapReduce(
                     function() { emit( this.cust_id, this.amount ); },
                     function(key, values) { return Array.sum( values ) },
                     {
                       query: { status: "A" },
                       out: "order_totals"
                     }
)
```

Collection
map
reduce
query
output
{ cust_id: "A123", amount: 500, status: "A" }
{ cust_id: "A123", amount: 250, status: "A" }
{ cust_id: "B212", amount: 200, status: "A" }
{ cust_id: "A123", amount: 300, status: "D" }
orders
query
{ cust_id: "A123", amount: 500, status: "A" }
{ cust_id: "A123", amount: 250, status: "A" }
{ cust_id: "B212", amount: 200, status: "A" }
map
{ "A123": [500, 250] }
{ "B212": 200 }
reduce
{ _id: "A123", value: 750 }
{ _id: "B212", value: 200 }
order_totals

Fig. 15: MapReduce using Mongo query

- Indexing: Indexes are special data structures that store a small portion of the collection's data set in an easy to traverse form[10]. Fields in a MongoDB document can be indexed with primary and secondary indices. The following diagram illustrates a query that selects and orders the matching documents using an index:



```
db.users.find( { score: { "$lt": 30 } } ).sort( { score: -1 } )
```

Fig. 16: Indexing[10]

- Sharding: Sharding is a process to distribute the data across multiple machines to keep balance with rapid growth of data.

MongoDB uses replica sets consisting two or more copies of data which makes it more scalable and reliable.

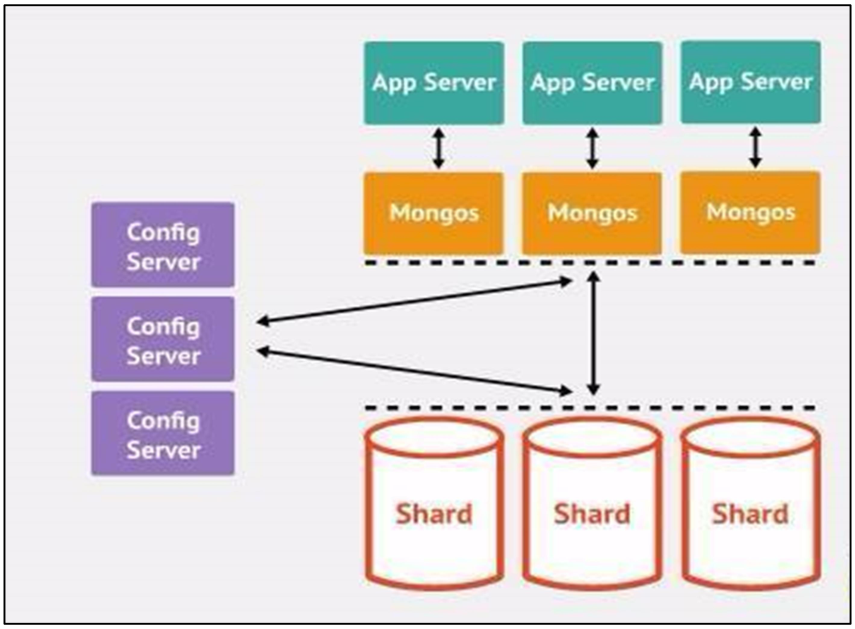
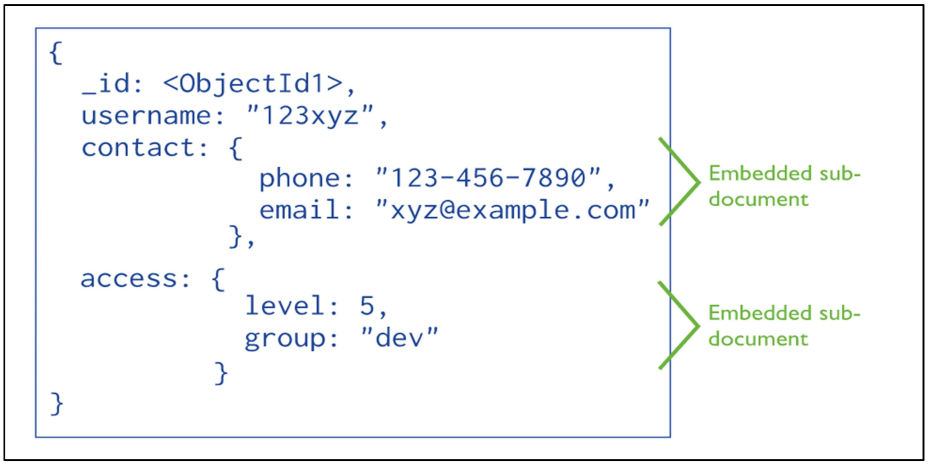


Fig. 17: Sharding[11]

- Document Structure: MongoDB allows related data to be embedded within a single document. These denormalized data models allow applications to retrieve and manipulate related data in a single database operation[12]. The following example shows how related documents are embedded within a single document:



```
{
  _id: <ObjectId1>,
  username: "123xyz",
  contact: {
              phone: "123-456-7890",
              email: "xyz@example.com"
            },
  access: {
             level: 5,
             group: "dev"
           }
}
```

Fig. 18: Embedding data[12]

- Capped collection: MongoDB supports fixed-size collections called capped collections[8]. It inserts data until the specified size has been reached. After that it behaves like a circular queue and starts storing data from the first empty space at front. The **db.createCollection** command is used to create a capped collection.
  db.createCollection('logs', {capped: true, size: 2097152}) this command limits the capped collection to 2 MB [11].

3.2. MongoDB Vs MySQL: Comparison between MongoDB and MySQL on various parameters are discussed below:

- Stored data: MongoDB stores each individual record as JSON-like documents in key-value pair format whereas, in MySQL each record is stored as row in the table.
- Flexibility: MongoDB is a NoSQL database and it provides dynamic schema. That means the documents of the same collection may differ by the structure as the structure of the incoming data does not need to be defined earlier. On the other hand, in MySQL the structure of the schema must be determined and declared before inserting the data as it uses Structured Query Language. Once the structure of the schema is defined, it cannot be changed. Therefore, MongoDB is more flexible that MySQL.
- Performance: In case of large data MySQL is quite slower than MongoDB to perform the operations. The following graph shows the comparison of performances between MySQL and MongoDB while operating on the same data.

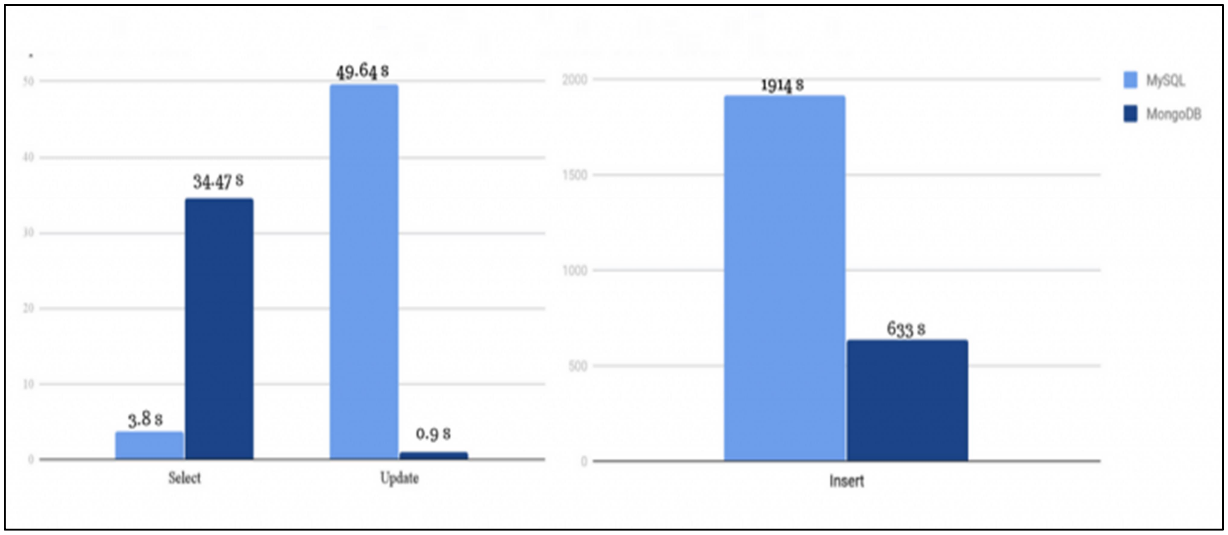


Fig. 19: Comparison of performances between MongoDB and MySQL[13]

- Scalability: MongoDB provides new levels of availability and scalability due to the use of sharding method which is unachievable with MySQL.
- Risk: In MySQL the risk of SQL injection attacks whereas, MongoDB is less risky in case of attack due to design.
- Analysis: MySQL should be used when the data is not large and structured and when the traditional relational database operations can be performed. But MongoDB has the potential to maintain and store structured as well as unstructured data with rapid growth.

From the above comparisons, it is clear that as data is growing faster than ever before, MongoDB would be a great choice rather than RDBMS.

Chapter Four: Windows Log

4.1. Categories of windows event logs:

Windows logs are broadly classified into six categories:

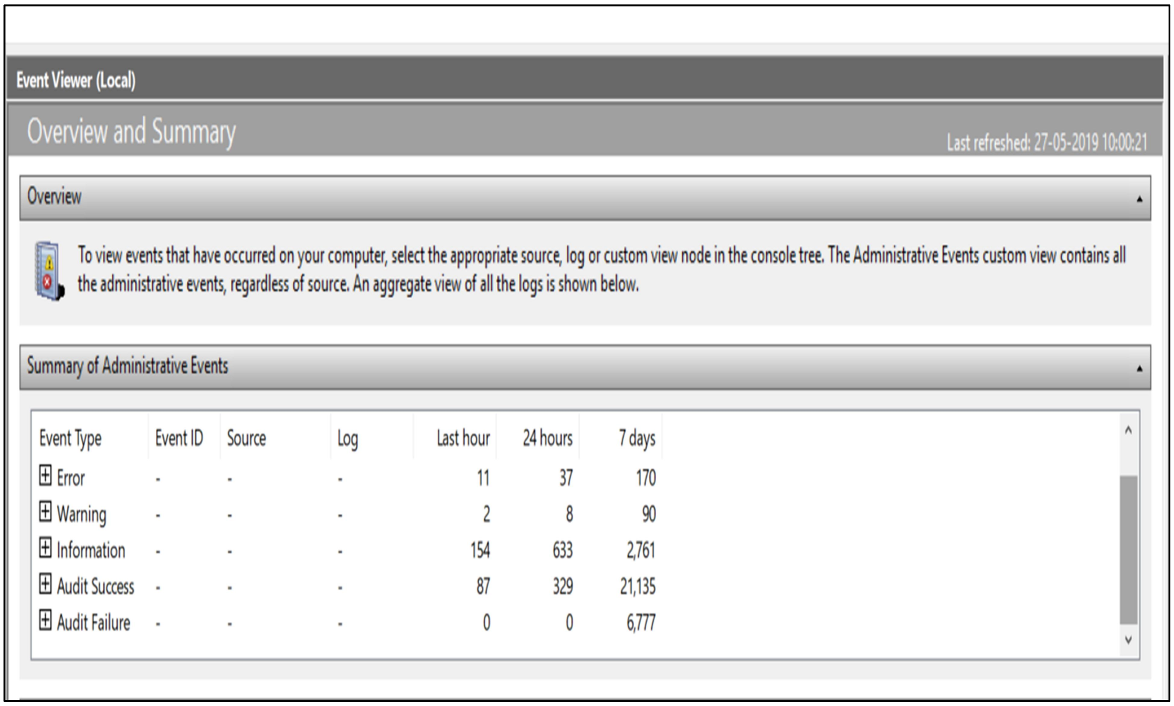
Event Log Type	Description
Application Log	When an application is started, an event is logged under this type of log and any event occurred for the running application is logged. These are determined by the developers while developing the application. Eg.: When a service is started or when the initialization process is failed for an application, all those information gets recorded in Application Log.
System Log	Whatever event is occurred in Operating System is logged in System log. Eg.: Failure to start a drive during startup is logged under System Logs
Security Log	Any event related to the security of the system. Eg.: valid and invalid Logins and logoffs, any event related to an object etc. are logged under this category.
Directory Service Log	Records events of AD. This log is accessible only on domain controllers.
DNS Server Log	Records events for DNS servers and name resolutions. This log is available only for DNS servers.
File Replication Service Log	Records events of domain controller replication. This log is available only on domain controllers[12].

4.2. <u>Types of events:</u> Each event entry is classified by type that measures the severity of the event. Each event has its own level value indicating its severity level type. Each event must be of a single type that means no event having same id can have different event types. The five types of events are –

Event Type	Description
Error	When there is an event occurred due to a significant problem such as loss of data or loss of functionality, the event will have this level. For example, if a service fails to load during start-up or if the license activation for an application fails, an Error event is logged.
Warning	An event that indicates some possible problems that may arise but for that moment it is not necessarily significant. When there is a warning event, it is possible for an application to recover from that phase of the event without any loss of data or functionality. For example, when a driver fails to load for a device, a Warning event is logged.
Information	When a task is completed successfully such as an application, driver, or service, an information event is logged. For example, when a service is started successfully, it is considered as an Information event. Or when the operating system started as well as is shutting down the event is logged along with the system time as an Information event.
Success Audit	An event that is related to a successful security access attempt is considered as a Success Audit. It indicates that the task is completed successfully and no need to monitor the event. This event can be found in security log only. For example, if a new process is created

	successfully, the event is logged along with the creator details and process information as a Success Audit event.
Failure Audit	An event that records a failed attempt to an audited security access. For example, if a user attempts to log on a system and fails, the attempt is logged as a Failure Audit event.

The Event Viewer lists these types as below –



Event Viewer (Local)

Overview and Summary

Last refreshed: 27-05-2019 10:00:21

Overview

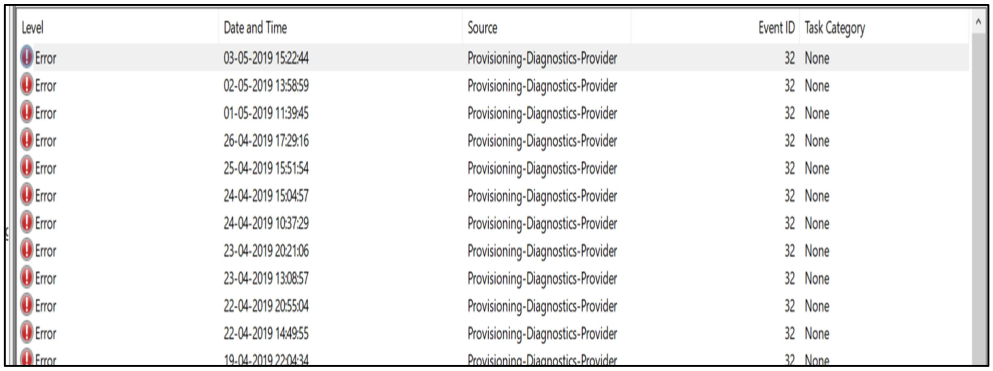
To view events that have occurred on your computer, select the appropriate source, log or custom view node in the console tree. The Administrative Events custom view contains all the administrative events, regardless of source. An aggregate view of all the logs is shown below.

Summary of Administrative Events

Event Type	Event ID	Source	Log	Last hour	24 hours	7 days
⊞ Error	-	-	-	11	37	170
⊞ Warning	-	-	-	2	8	90
⊞ Information	-	-	-	154	633	2,761
⊞ Audit Success	-	-	-	87	329	21,135
⊞ Audit Failure	-	-	-	0	0	6,777

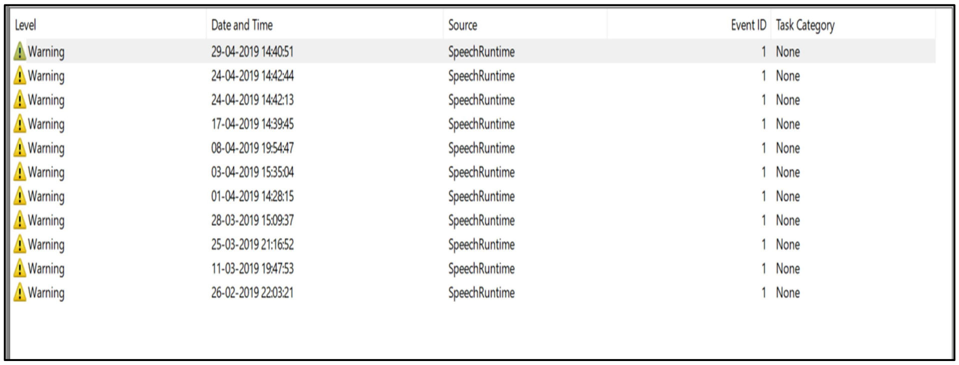
Fig. 20: Summary of event types

Each type is displayed with different icon in the list view of event log:



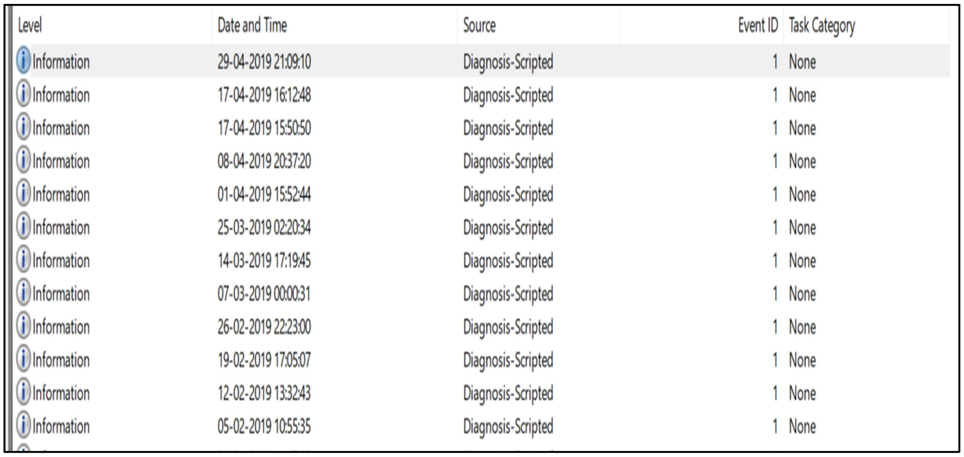
Level	Date and Time	Source	Event ID	Task Category
Error	03-05-2019 15:22:44	Provisioning-Diagnostics-Provider	32	None
Error	02-05-2019 13:58:59	Provisioning-Diagnostics-Provider	32	None
Error	01-05-2019 11:39:45	Provisioning-Diagnostics-Provider	32	None
Error	26-04-2019 17:29:16	Provisioning-Diagnostics-Provider	32	None
Error	25-04-2019 15:51:54	Provisioning-Diagnostics-Provider	32	None
Error	24-04-2019 15:04:57	Provisioning-Diagnostics-Provider	32	None
Error	24-04-2019 10:37:29	Provisioning-Diagnostics-Provider	32	None
Error	23-04-2019 20:21:06	Provisioning-Diagnostics-Provider	32	None
Error	23-04-2019 13:08:57	Provisioning-Diagnostics-Provider	32	None
Error	22-04-2019 20:55:04	Provisioning-Diagnostics-Provider	32	None
Error	22-04-2019 14:49:55	Provisioning-Diagnostics-Provider	32	None
Error	19-04-2019 22:04:34	Provisioning-Diagnostics-Provider	32	None

Fig. 21: Event type: Error



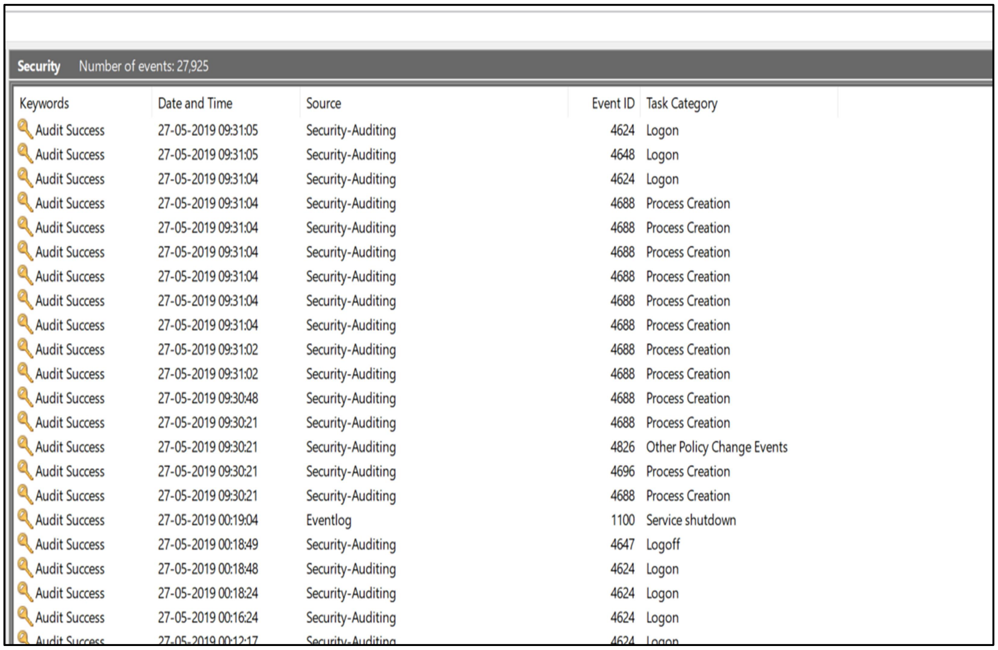
Level	Date and Time	Source	Event ID	Task Category
Warning	29-04-2019 14:40:51	SpeechRuntime	1	None
Warning	24-04-2019 14:42:44	SpeechRuntime	1	None
Warning	24-04-2019 14:42:13	SpeechRuntime	1	None
Warning	17-04-2019 14:39:45	SpeechRuntime	1	None
Warning	08-04-2019 19:54:47	SpeechRuntime	1	None
Warning	03-04-2019 15:35:04	SpeechRuntime	1	None
Warning	01-04-2019 14:28:15	SpeechRuntime	1	None
Warning	28-03-2019 15:09:37	SpeechRuntime	1	None
Warning	25-03-2019 21:16:52	SpeechRuntime	1	None
Warning	11-03-2019 19:47:53	SpeechRuntime	1	None
Warning	26-02-2019 22:03:21	SpeechRuntime	1	None

Fig. 22: Event type: Warning



Level	Date and Time	Source	Event ID	Task Category
Information	29-04-2019 21:09:10	Diagnosis-Scripted	1	None
Information	17-04-2019 16:12:48	Diagnosis-Scripted	1	None
Information	17-04-2019 15:50:50	Diagnosis-Scripted	1	None
Information	08-04-2019 20:37:20	Diagnosis-Scripted	1	None
Information	01-04-2019 15:52:44	Diagnosis-Scripted	1	None
Information	25-03-2019 02:20:34	Diagnosis-Scripted	1	None
Information	14-03-2019 17:19:45	Diagnosis-Scripted	1	None
Information	07-03-2019 00:00:31	Diagnosis-Scripted	1	None
Information	26-02-2019 22:23:00	Diagnosis-Scripted	1	None
Information	19-02-2019 17:05:07	Diagnosis-Scripted	1	None
Information	12-02-2019 13:32:43	Diagnosis-Scripted	1	None
Information	05-02-2019 10:55:35	Diagnosis-Scripted	1	None

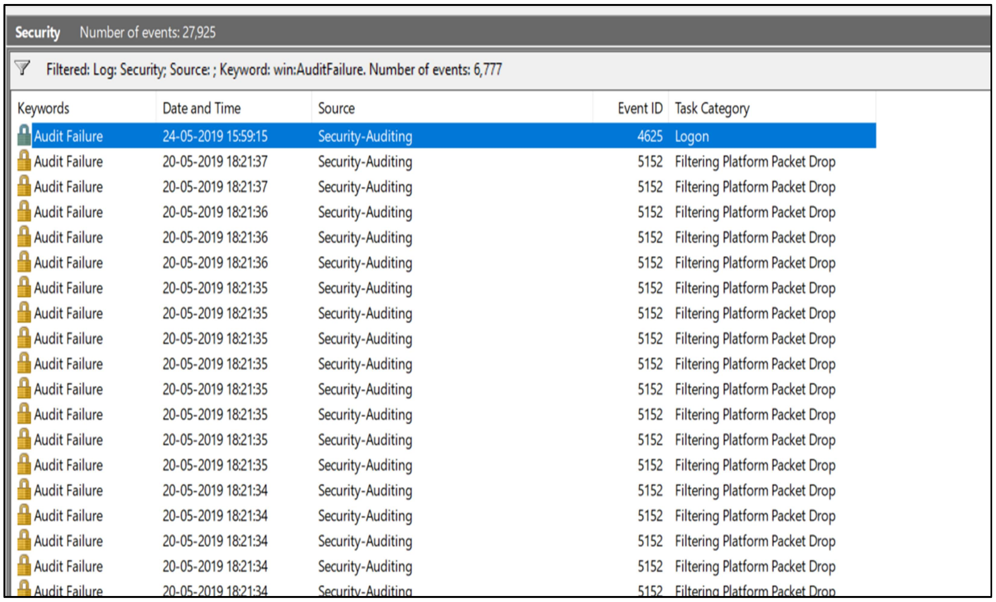
Fig. 23: Event type: Information



Security Number of events: 27,925

Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	27-05-2019 09:31:05	Security-Auditing	4624	Logon
Audit Success	27-05-2019 09:31:05	Security-Auditing	4648	Logon
Audit Success	27-05-2019 09:31:04	Security-Auditing	4624	Logon
Audit Success	27-05-2019 09:31:04	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:31:04	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:31:04	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:31:04	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:31:04	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:31:04	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:31:02	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:31:02	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:30:48	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:30:21	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 09:30:21	Security-Auditing	4826	Other Policy Change Events
Audit Success	27-05-2019 09:30:21	Security-Auditing	4696	Process Creation
Audit Success	27-05-2019 09:30:21	Security-Auditing	4688	Process Creation
Audit Success	27-05-2019 00:19:04	Eventlog	1100	Service shutdown
Audit Success	27-05-2019 00:18:49	Security-Auditing	4647	Logoff
Audit Success	27-05-2019 00:18:48	Security-Auditing	4624	Logon
Audit Success	27-05-2019 00:18:24	Security-Auditing	4624	Logon
Audit Success	27-05-2019 00:16:24	Security-Auditing	4624	Logon
Audit Success	27-05-2019 00:12:17	Security-Auditing	4624	Logon

Fig. 24: Event type: Audit Success



Security Number of events: 27,925

Filtered: Log: Security; Source: ; Keyword: win:AuditFailure. Number of events: 6,777

Keywords	Date and Time	Source	Event ID	Task Category
Audit Failure	24-05-2019 15:59:15	Security-Auditing	4625	Logon
Audit Failure	20-05-2019 18:21:37	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:37	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:36	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:36	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:36	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:35	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:35	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:35	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:35	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:35	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:35	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:35	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:35	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:34	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:34	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:34	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:34	Security-Auditing	5152	Filtering Platform Packet Drop
Audit Failure	20-05-2019 18:21:34	Security-Auditing	5152	Filtering Platform Packet Drop

Fig. 25: Event type: Audit Failure

4.3. Description of fields:

If event log data is converted into CSV format, each event in a log entry contains the following 27 fields

Fields	**Description**
Message	The detailed description of the event written in text format.
Id	The event identifier used to uniquely identify each event. The value is specific to the event source for the event.
version	The version number of the event's definition.
Qualifiers	First 16bits of EventId
level	Indicates severity level of event. The levels are: Critical, Error, Warning, Information, Verbose.
Task	In case of group events, task is the major component (e.g. networking or database) of the provider which is performing.
Opcode	Used to identify the operation that is being performed by the component.
Keywords	Used to classify different types of events
RecordId	The record number of the event log record. First record is numbered as 1. After reaching maximum value it starts from 0.
ProviderName	The process that wrote the event record.
ProviderId	Corresponding Id to the ProviderName.
LogName	The name of the event log from which this record was read. This value is one of the

	names from the event_logs collection in the configuration.
ProcessId	Identifies the process that generated the event.
ThreadId	Identifies the thread that generated the event.
MachineName	The name of the system on which the event occurred.
UserId	The security identifier (SID) of the user logged in during the occurrence of the event. It is given in string format.
TimeCreated	Identifies the date and time when the event was recorded. The time stamp includes either the SystemTime attribute or the RawTime attribute.
ActivityId	A globally unique identifier that identifies the current activity. The events that are published with this identifier are part of the same activity[15].
RelatedActivityId	A globally unique identifier that identifies the activity to which control was transferred to. The related events would then have this identifier as their ActivityID identifier[15].
ContainerLog	File path that contains the logged event.
MatchedQueryIds	

Bookmark	
LevelDisplayName	Level of the event corresponding to the level value.
OpcodeDisplayName	Name of operation corresponding to the opcode value.
TaskDisplayName	Name of task performed in the event.
KeywordsDisplayNames	
Properties	

4.4. <u>Severity Levels of event:</u> Levels are used to group events and typically indicate the severity or verbosity of an event.

Level Value	Level Name	Description
1	Critical	This level indicates the most severe problems. It means the system administrator needs to monitor the log immediately. Event ID 41 generated by Kernel-Power is an event of critical level where the system has rebooted without cleanly shutting down first.
2	Error	This level indicates some problems due to failure to operate the device expectedly but it does not demand immediate attention as it is not as severe as critical level.

		Event ID 8198 generated by Security-SPP is an event of error level where license activation failed.
3	Warning	This level indicates that a component or application is not in an ideal state and some further actions could arise significant problems. Event ID 1014 generated by DNS Client Events is an example of warning level where name resolution for a domain name timed out after none of the configured DNS servers responded.
4	Information	It indicates a noncritical event. It passes the information relevant to the event occurred. Event ID 50036 generated by Dhcp-Client is an informational event where it gives the message that DHCPv4 client service is started.
	Verbose	It indicates the verbose status, such as progress or success messages. Generally the verbose logging option is disabled as it generates more information than usual log files and thus it may cause slow performance of the system. When the detailed information is needed such as for troubleshooting, the verbose logging needs to be enabled.

Chapter Five: Log Analysis

5.1. What is log analysis: Log analysis is the term used for analysis of computer-generated records for helping organizations, businesses or networks in proactively and reactively mitigating different risks[16]. In an organization, log analysis plays an important role in security. Log analysis helps to collect the statistics about the activities done in the organization and analyse whether those are regular or unusual. If the log analysis job is maintained a regularity, the activities within the organization can be made more secure and efficient. Thus with the help of log analysis, risk can be reduced within an organization.

5.2. Importance of fields: Log analysis includes extracting information from each and every field of the log file, understanding it, and reaching to a conclusion whether to classify the logged data as safe or suspicious. All the fields are equally important in log analysis. Among them the mostly used fields are: Message, Id, TimeCreated, MachineName, Level, ProviderName etc.

5.2.1. Message: The message field describes the event's record in a user-friendly way. The first few lines contains a brief description in text format. This description can be useful while querying for some keywords from the message field. For example, if a user wants to check which services were started in the whole log file, a query can be made to find the keyword 'service' within the message. This can be done using MongoDB query as given below:

- db.collection_name.find({'Message':{$regex:'service', $options:'i'}})

This query matches all the records having 'service' in the Message field and shows all the fields of the matched records.

The Message field then contains the detailed description of the source and target of the event, if any. This description is divided into some subfields which shows different parameters and their corresponding values. These subfields vary with the events. As an example, a successful logon event's Message (4624) has subfields subject, logon information, impersonation level, new logon, process information, network information, and detailed authentication information.

- The Subject field indicates the account on the local system which requested the logon.
- The Logon Information indicates how the logon session was created. It has some parameters such as Logon Type which indicates the kind of logon (interactive, network, batch, service etc.) that occurred, Restricted Admin Mode that indicates whether the logon was interactive or a remote logon. Another parameter is Virtual Account which has two possible values 'yes' and 'no', and the last one is Elevated Token that indicates the permissions to the user's access.
- The Impersonation Level field indicates the extent to which a process in the logon session can impersonate.
- The New Logon fields indicate the account for whom the new logon was created. It contains the logon id and the other information about the account that is logged in.
- The Process Information includes the process id and the process name which was started due to occurrence of the event.
- Network Information field includes the user's current system details i.e. workstation name, port, IP address. It indicates where the user was while logging in. Mainly for remote logon session this field is useful.
- The authentication information fields provide detailed information about this specific logon request.

By analysing the Message field remote logon event can be detected from the database. The Mongo query to find out remote logon is given below:

- ➢ db.collection_name.find({'Message':{$regex:'Restricted Admin Mode:\tYes'}})

Similarly, when a process is created in the system, the Message field of the event (4688) has the subfields for the creator’s details, target’s details and the process information. For ease of understanding the functions of the subfields are described briefly in the Message field.

5.2.2. **Id:** Any particular event can be uniquely identified by the event id. Each event id is mapped to a descriptive format which is shown in the Message field. Event id is useful to find single activity as well as batched activity. A group of related activities can be clustered and subsequently retrieved with the help of event identifier and some other fields. As an example, to find all the failed logon events only the event id is enough for the query:

- ➢ db.collection_name.find({‘Id’:4625})

4625 is the event id for logon failure.

While accessing an object in the system, a series of events are generated one by one. To group these events the following steps can be followed:

1. Find event id 4656 that indicates a handle to an object was requested.
2. Find event id 4663 where an attempt was made to access an object.
3. Match the handle id with the same of the event mentioned step 1.
4. Match the timestamp sequentially and select the event 4663 having same handle id of event 4656.
5. Read the Message field of the 4663 event and get the value of parameter Accesses under the Access Request Information subfield. This value informs about the type of access request i.e. read or write.

6. Find event 4658 where the handle to the object was closed and match the handle id.

5.2.3. TimeCreated: This is the most important and useful field in log analysis. Generally the time of the event is stored in the system time format. As the format can vary system to system, the time should be normalized in a specific unit to arrange the data. The logs can be sorted by the normalized time. In MongoDB, there exists 'datetime' object to represent dates and times in MongoDB documents. With the help of this field various conclusions can be derived. Such as if any internal attack occurred in the organization, it should be checked which users were logged in through which machine during the time of attack. This analysis can be done using the following steps:

1. Filter the data by matching the event id with 4624 (logon) or 4634 (logoff).
2. Sort the filtered log in either ascending or descending order by time.
3. As the name of user is given as the value of Account Name in the message field, check the number of logon and logoff events for each user.
4. If there are more number of logon events than logoff for a user, then flag those events.
5. From the flagged events retrieve all distinct user name.

5.2.4. **MachineName:** The MachineName field reflects the source machine from where the event was generated. If any event is found suspicious, the source of the event can be detected from this field.

5.2.5. Level: From the level of the event, one can understand how severe the event is. If the frequency of error or warning events is noticed as very high, then the administrator should take immediate action to monitor the logs to prevent the organization from possible attack or damage. The following query returns the total number of events of warning type:

- ➢ db.collection_name.find({‘Level’ : 3}).count()

5.2.6. <u>ProviderName:</u> This field indicates the source process of the event. To filter the data by particular source, query can be run using this field. As an example, a query to get all the events created by MsiInstaller:

- ➢ db.collection_name.find({‘ProviderName’:’MsiInstaller’})

Thus any event related to the installer can be fetched.

To reduce risk, the other fields are also useful in log analysis.

Chapter Six: Implementation

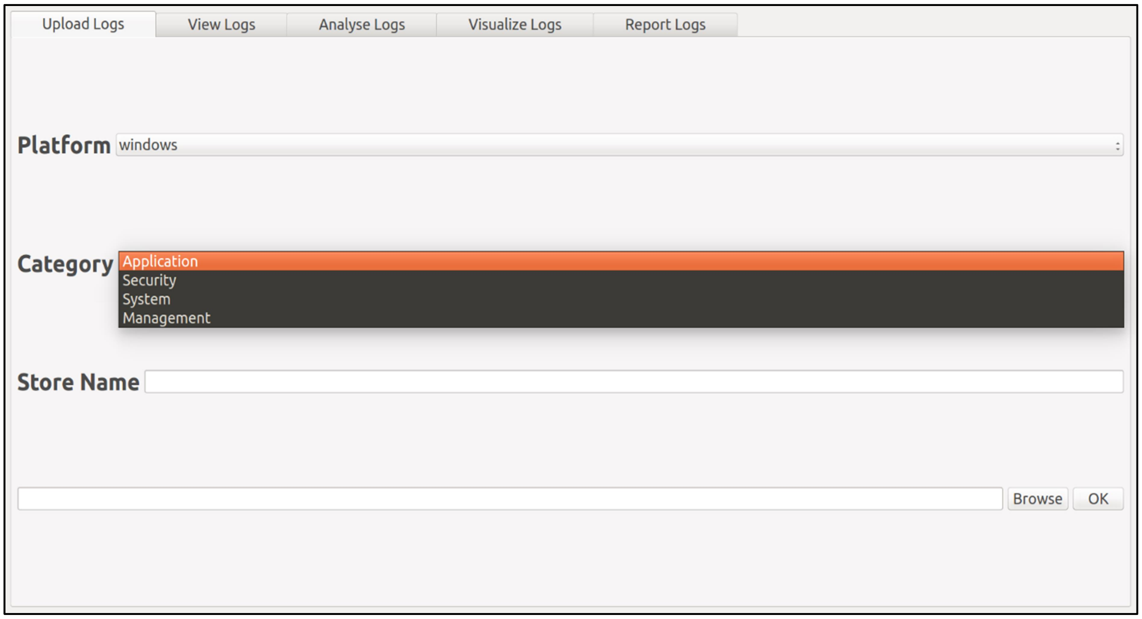
6.1. Overview of the tool: It is difficult to analyse log data of large scale manually. The tool is being implemented for log analysis purposes. It helps to extract important data from the log files and if any irregularity is found, it generates alert so that the user can be aware of that activity without analysing it manually. For now it is implemented for MS Windows logs only, the features of syslog can be added later to extend the functionality of the tool. Visualisation feature has been added to depict various kinds of activity present in the log and based on those activities report will be generated after prolonged analysis.

There are five tabs implemented for different steps in deriving useful metrics:

- Upload Logs
- View Logs
- Analyse Logs
- Visualize Logs
- Report

The features of each tabs are described below.

6.1.1. Upload: This tab allows the user to upload the log file to be stored and monitored later. It categorises the logs into two types based on the platform: Windows and Linux. To upload the log, at first the platform should be chosen. This categorisation is done for selecting the parser implemented as the formats of Windows log and Linux Log are different in terms of their fields. On the basis of the platform, the log is subcategorised under each platform. For Windows, four types of options are provided: Application, Security, System, and Management whereas, under Linux the subcategories are Application, Event, System, and Service. Instead of storing each file in different table, a centralised storage is applied in this tool. For each platform only a single collection of MongoDB will be used to store all files of every category and to distinguish the files, a field will be used named ‘Store Name’. Every time the user uploads a log file, a unique store name must be provided and in case of repetition of store name, a message will be generated saying to provide a unique name. At first the raw event file (.evtx) has to be converted in CSV format and then the CSV file should be chosen to be stored. In case of non-existing file path, or unsupported file format, an error message will be generated with a message. After selecting the CSV file the parser started to be executed at the back end. The parser reads the file and stores the twenty seven fields along with the mentioned store name and the default primary key in the database.



Platform windows

Category Application

Store Name

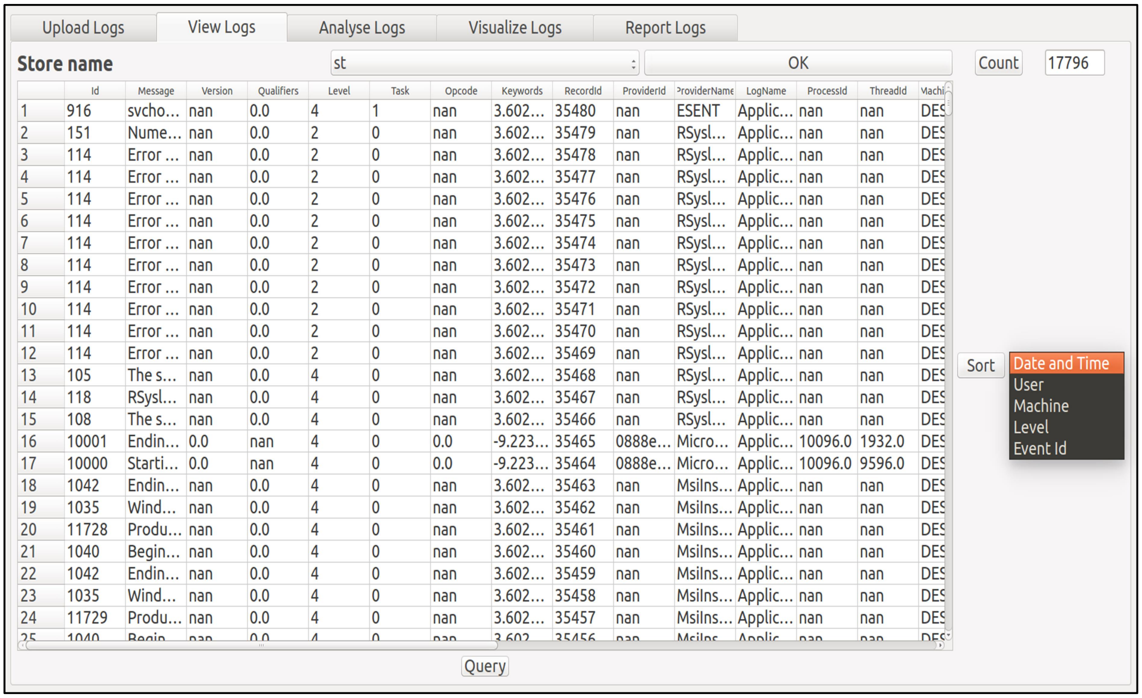
Fig. 26: Screenshot of Upload tab

6.1.2. View: A dropdown box is added in the view tab to select a particular store name. After selecting the store, the control fetches the data having the selected store name from the database and shows the log data in tabular form. A button named ‘Count’ will show the total count of the data shown in the table after being clicked. A feature is also added to sort the data in the table by the following fields:

- Date and time

- User
- EventId
- Level
- Machine name

The data filtration feature has been provided through the Query button. After being clicked, it opens another window that allows to enter the values of all the important fields. The values are passed through the query and shows the retrieved matched data in the table of the view tab. With the help of this feature, the user can explore the data based on customised log properties. This feature provides ease of access the data with expected field values.



Store name: st

Count: 17796

	Id	Message	Version	Qualifiers	Level	Task	Opcode	Keywords	RecordId	ProviderId	ProviderName	LogName	ProcessId	ThreadId	Machi
1	916	svcho...	nan	0.0	4	1	nan	3.602...	35480	nan	ESENT	Applic...	nan	nan	DES
2	151	Nume...	nan	0.0	2	0	nan	3.602...	35479	nan	RSysl...	Applic...	nan	nan	DES
3	114	Error ...	nan	0.0	2	0	nan	3.602...	35478	nan	RSysl...	Applic...	nan	nan	DES
4	114	Error ...	nan	0.0	2	0	nan	3.602...	35477	nan	RSysl...	Applic...	nan	nan	DES
5	114	Error ...	nan	0.0	2	0	nan	3.602...	35476	nan	RSysl...	Applic...	nan	nan	DES
6	114	Error ...	nan	0.0	2	0	nan	3.602...	35475	nan	RSysl...	Applic...	nan	nan	DES
7	114	Error ...	nan	0.0	2	0	nan	3.602...	35474	nan	RSysl...	Applic...	nan	nan	DES
8	114	Error ...	nan	0.0	2	0	nan	3.602...	35473	nan	RSysl...	Applic...	nan	nan	DES
9	114	Error ...	nan	0.0	2	0	nan	3.602...	35472	nan	RSysl...	Applic...	nan	nan	DES
10	114	Error ...	nan	0.0	2	0	nan	3.602...	35471	nan	RSysl...	Applic...	nan	nan	DES
11	114	Error ...	nan	0.0	2	0	nan	3.602...	35470	nan	RSysl...	Applic...	nan	nan	DES
12	114	Error ...	nan	0.0	2	0	nan	3.602...	35469	nan	RSysl...	Applic...	nan	nan	DES
13	105	The s...	nan	0.0	4	0	nan	3.602...	35468	nan	RSysl...	Applic...	nan	nan	DES
14	118	RSysl...	nan	0.0	4	0	nan	3.602...	35467	nan	RSysl...	Applic...	nan	nan	DES
15	108	The s...	nan	0.0	4	0	nan	3.602...	35466	nan	RSysl...	Applic...	nan	nan	DES
16	10001	Endin...	0.0	nan	4	0	0.0	-9.223...	35465	0888e...	Micro...	Applic...	10096.0	1932.0	DES
17	10000	Starti...	0.0	nan	4	0	0.0	-9.223...	35464	0888e...	Micro...	Applic...	10096.0	9596.0	DES
18	1042	Endin...	nan	0.0	4	0	nan	3.602...	35463	nan	MsiIns...	Applic...	nan	nan	DES
19	1035	Wind...	nan	0.0	4	0	nan	3.602...	35462	nan	MsiIns...	Applic...	nan	nan	DES
20	11728	Produ...	nan	0.0	4	0	nan	3.602...	35461	nan	MsiIns...	Applic...	nan	nan	DES
21	1040	Begin...	nan	0.0	4	0	nan	3.602...	35460	nan	MsiIns...	Applic...	nan	nan	DES
22	1042	Endin...	nan	0.0	4	0	nan	3.602...	35459	nan	MsiIns...	Applic...	nan	nan	DES
23	1035	Wind...	nan	0.0	4	0	nan	3.602...	35458	nan	MsiIns...	Applic...	nan	nan	DES
24	11729	Produ...	nan	0.0	4	0	nan	3.602...	35457	nan	MsiIns...	Applic...	nan	nan	DES
25	1040	Begin...	nan	0.0	4	0	nan	3.602...	35456	nan	MsiIns...	Applic...	nan	nan	DES

Fig. 27: Screenshot of View tab

6.1.3. <u>Analyse:</u> When a significant damage or any suspicious activity occurs in an organization, it is necessary to check the active users and the corresponding machines at the time when the event occurred. This makes the diagnosis to be more properly. The analysis module introduces a feature which selects and shows the list of active machines and users at a particular timestamp. There will be a field to enter any date and time in the 'DD-MM-YYYY HH:mm:ss' format and the calculation will be processed at the back-end to find out which user was logged in at that time in which system. If the user id is missing in the log data, then only the machine name will be shown in the list. Another outcome of log analysis process is generating alerts if any suspicious activity is found. This feature is also added in this module where a pre-defined rule set is provided. Whenever any event violates the rule set, it will be marked as suspicious and an alert will be generated with the details of the marked event. The rule set is given below:

- The audit log must not be cleared
- The system audit policy must not be changed
- Changing permissions on an object is not allowed
- Remote logon is not allowed

To check whether the activities maintained the above rules, the event id plays the main role. The events are checked one by one and are matched with the set of the event id which violates any of the above rules. If a match is found, alert is generated.

The module also checks suspicious activity using batch events. Such as:

- Generates alert if frequency of failed logon event to the same account exceeds three times within five minutes.
- If a user's account was locked out after repeated logon failures.
- Generates alert if an attempt to open an object was failed.
- When an object was deleted.
- When an attempt (both read and write) was made to access an object.

The last three scenarios, related to handle to an object, helps to maintain the CIA (Confidentiality, Integrity, and Availability) triad for any object. To make sure these events are generated in the system, the audit policy should be configured as well as the permissions to individual files should also be defined before.

6.1.4. Visualize: To ease understanding of large data various charts can be used. Sometimes the number of events in a small time duration increases in a surprisingly high rate which can be considered an unusual scenario under an organization. To visualize the change in count of events over time, a chart is implemented in this module. As it is shown earlier that the severity levels play a major role to detect the possibility of being suspected of the event, it should be monitored that how frequently several types of level are being occurred. For that job, a pie-chart is provided where all the levels are shown with their frequencies as numerical proportion. As a result, if it is found that the proportion of critical or error level is high in an unusual proportion, then it indicates that some activities should be monitored before an attack occurs.

6.1.5. Report: The events flagged as suspicious in the analyze module will be summarized in a pdf document which can be seen from the report tab.

6.2. Technologies used: For implementing the tool, Python 2.7 has been used to develop the front-end as well as back-end and MongoDB has been used for storing data. As MongoDB supports unstructured data with a large volume, MongoDB has been chosen over relational database.

6.3. **Tool Architecture:** The flow of the modules of the tool is depicted below:

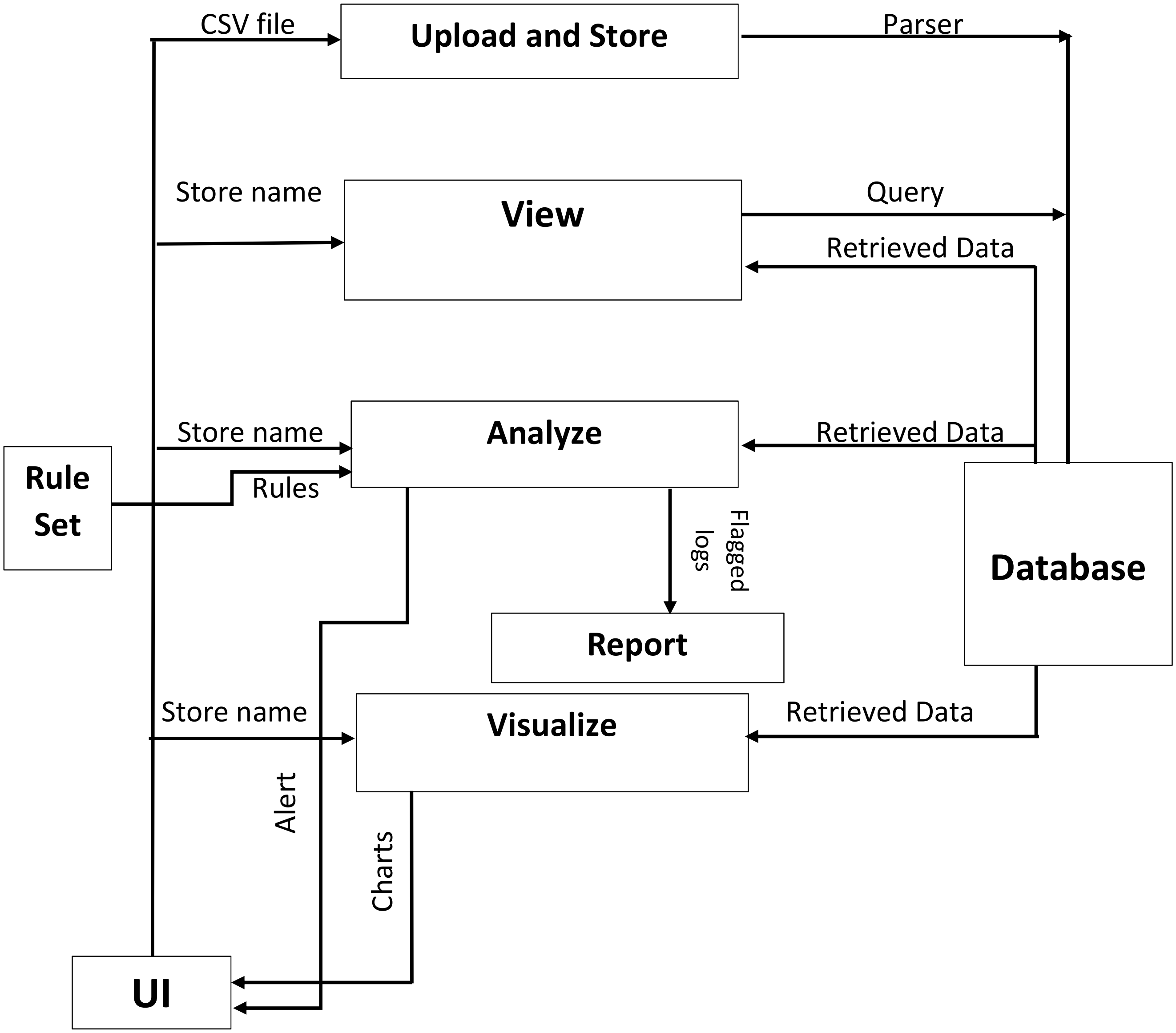


Fig. 28: Tool architecture

Chapter Seven

7.1. Future work:

In this project, some of the rules were implemented to analyze the log and detect attempt to attack. Addition of rules to the rule set can be done to make the product more efficient to find more possible threats for an organization. By correlating multiple types of logs the activities of individual users can be tracked sequentially. This feature will make the analysis more effective and efficient.

To understand the log more properly the visualization module can be modified by adding various types of charts to plot different fields with time.

In this project the input is in CSV format. To extend the functionality of the product a feature can be added that takes the raw file (file with evtx extension) as input and converts it into CSV as intermediate file.

Availability of some complex queries will make the product more user friendly. In the table an option to hide any property can be added to increase the flexibility of the product.

Moreover, real-time streaming of data can be the most powerful feature of the tool. To use the tool in a broad way, acceptance of logs from all type of sources can be allowed by developing different parsers for different sources.

7.2. Conclusion:

It can be concluded that this tool can analyze the logs based on a rule set. Modification to the rule set can only be done by the developer. As any prediction has not been made throughout the whole algorithm, the authenticity of the output can be expected to be accurate. Addition of advanced rules will enhance the efficiency of the tool.

References:

1. https://en.wikipedia.org/wiki/Log_file
2. https://onlinelibrary.wiley.com/doi/full/10.1002/sec.1677
3. https://docs.microsoft.com/en-us/windows-hardware/drivers/kernel/overview-of-windows-components
4. https://docs.microsoft.com/en-us/windows-hardware/drivers/gettingstarted/user-mode-and-kernel-mode
5. https://searchwindowsserver.techtarget.com/definition/Windows-event-log
6. https://docs.microsoft.com/en-us/windows/desktop/secauthz/audit-generation
7. https://en.wikipedia.org/wiki/Event_Viewer
8. https://en.wikipedia.org/wiki/MongoDB
9. https://docs.mongodb.com/manual/aggregation/

10. https://docs.mongodb.com/manual/indexes/

11. https://www.tutorialsjar.com/key-features-of-mongodb/

12. https://docs.mongodb.com/manual/core/data-modeling-introduction/

13. https://dzone.com/articles/comparing-mongodb-amp-mysql

14. https://www.manageengine.com/network-monitoring/Eventlog_Tutorial_Part_I.html

15.https://www.elastic.co/guide/en/beats/winlogbeat/5.1/exported-fields-eventlog.html

16. https://www.techopedia.com/definition/31756/log-analysis

17.https://www.manageengine.com/products/active-directory-audit/help/getting-started/windows-workstations-advanced-audit-policy.html

18.https://www.ultimatewindowssecurity.com/securitylog/book/page.aspx?spid=chapter2

19. https://www.ittsystems.com/best-event-log-analysis-tools/

20. https://www.solarwinds.com/log-analyzer

21. https://www.keycdn.com/blog/log-analysis-tools

22. https://www.loggly.com/docs/using-loggly/

23.https://www.manageengine.com/products/eventlog/help/StandaloneManagedServer-UserGuide/Introduction/about-eventlog-analyzer.html

24.https://www.elastic.co/guide/en/logstash/current/introduction.html#power-of-logstash