



Logging Task

[Visit our website](#)

Introduction

Logging is an essential aspect of cyber security that involves the collection and recording of data about various activities within a computer system or network. This data can be stored within a database or can be static text files depending on the system. It is also typical to see logs forwarded to a Syslog server that aggregates logs from many systems. This data can be used to monitor and detect potential security threats, as well as track user behaviour and identify any unauthorised access or activity.

Access control and **authentication** are critical components of logging, as they help to ensure that only authorised users can access and modify sensitive data. **Network security** and **defence** involve protecting against external threats such as hackers and malware. In contrast, **user accountability** and **responsibility** ensure that users are held accountable for their actions within the system.

Understanding the purpose of logging

Logging has several important functions.

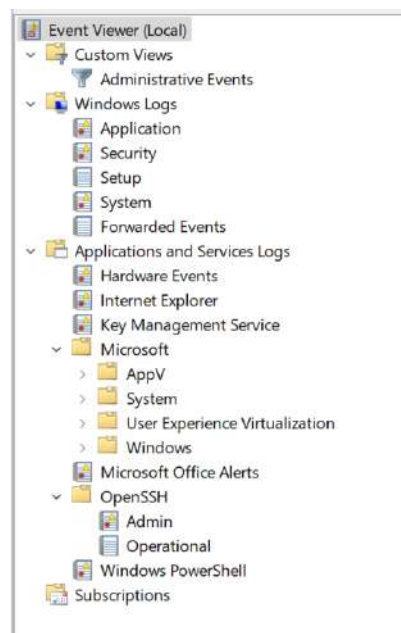
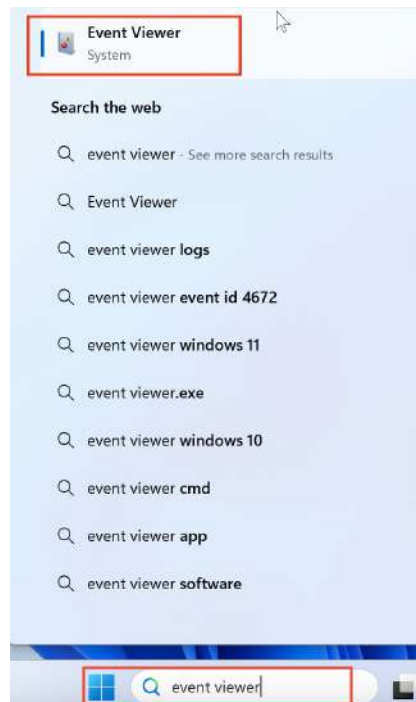
1. **Security:** Logging can help to detect and prevent security threats by providing a record of activity that can be used to identify unusual or suspicious behaviour. This can include failed login attempts, access to sensitive data, and detected security threats. **Security Information and Event Management (SIEM)** solutions 'learn' the baseline behaviour of users and organisations so that deviations from this norm can be reported. For example, if a user logs in from home shortly after their access card is used for logging into an office three hours away, then a SIEM solution would detect this and alert the appropriate person for a response.
2. **Compliance:** Many industries have regulations that require organisations to maintain detailed logs of user activity and system events. These logs can be used to demonstrate compliance with these regulations, as well as track user behaviour and patterns of activity.
3. **Troubleshooting:** Logs can be used to troubleshoot problems with a system or network, including identifying the root cause of errors or failures. This can help to minimise downtime and improve the overall reliability of the system.
4. **Performance monitoring:** Logs can provide valuable insights into the operation and performance of a system or network, including data on usage patterns and resource consumption. This can help to optimise system performance and identify potential issues before they become serious problems.

Identifying types of logs

Different types of logging are used for different operational purposes and can often be used to detect security threats.

1. **System logs** contain information about the operation and performance of a computer system or network, including 'startup' and 'shutdown' events, system errors, and hardware and software changes. System logs can troubleshoot problems, optimise system performance, and detect security threats.
2. **Application logs** contain information about the operation and performance of specific applications, including errors, warning messages, and user activity. Application logs can be used to troubleshoot problems with individual applications, as well as to monitor user behaviour and identify potential security threats.
3. **Security logs** contain information about security-related events, such as failed login attempts, access to sensitive data, and detected security threats. Security logs can be used to monitor and detect potential security threats, track user activity and identify any unauthorised access. **Note:** It is important to note that security logs are not the only relevant logs for cyber security and it may be necessary to review other log types during or after a cyber security incident.
4. **Audit logs** contain detailed information about user activity, including the time, date, and user associated with specific actions. Audit logs can be used to track user behaviour, identify activity patterns, and ensure compliance with company policies and regulations.
5. **Network logs** contain information about network traffic and activity, including the source and destination of data, as well as the protocols and ports used. Network logs can monitor network performance, detect security threats, and optimise network configuration.
6. **Access logs** contain information about user access to a system or network, including the time, date, and user associated with specific access attempts. Access logs can be used to track user activity and identify patterns of activity, as well as to detect and prevent unauthorised access.

Here is a screenshot of the default log types on a Windows PC, which is similar to Windows Server. To navigate to this, search for “Event viewer” on the search bar, and click on the system tool as shown below.



For Ubuntu Linux, you can navigate to “Show Applications” on the bottom left and search for “Logs”.

Logging data considerations

Data sovereignty, retention, and storage

Data sovereignty refers to the legal jurisdiction under which data is collected, processed, and stored. It is an essential consideration in the context of logging, as different countries and regions may have differing laws and regulations governing data collection, use, and storage.

Data retention refers to the length of time that data is kept or stored. In the context of logging, data retention policies dictate how long logs must be maintained and may also specify which types of logs should be retained, for how long, and which may be deleted.

Data storage refers to the physical location where data is stored. In the context of logging, data storage considerations may include the type of storage media used, the security measures in place to protect the data and the availability and accessibility of the data.

All these factors can have an impact on the logging process, and it is essential for organisations to carefully consider their data sovereignty, retention, and storage policies when implementing logging systems. Failing to adequately address these issues can result in compliance issues, security vulnerabilities, and other problems.

Data integrity and security

Data integrity refers to the accuracy and consistency of data and is an essential consideration in the context of logging. Logs are often used as a source of evidence or proof of certain events or activities and must be accurate and reliable. To maintain data integrity, organisations may need to implement measures such as data backup and recovery and controls to prevent unauthorised access or modification of log data. This can help ensure that logs are accurate and trustworthy and can be used as a reliable source of evidence.

Data security is also essential in the context of logging, as logs often contain sensitive information on topics including user activity, system events, and security-related events. To protect this data, organisations may need to implement encryption, access controls, and physical security measures to prevent unauthorised access or tampering with log data.

Logging and cyber security

Security operations centre (SOC)

A security operations centre (SOC) is a dedicated team or unit within an organisation responsible for monitoring and analysing data from various systems and networks to detect and respond to security threats. The SOC plays a critical role in the real-world application of the topics covered in this task, as it is responsible for implementing the policies, processes, and technologies needed to protect against cyber threats.

Auditing and monitoring are critical activities within a SOC, as they involve the continuous review and analysis of data from various sources to identify potential security incidents. This can include reviewing logs, analysing network traffic, and scanning for anomalies or unusual behaviour using automated tools.

Event tracking and analysis involve identifying and analysing specific security-related events, such as failed login attempts or access to sensitive data. This can help to identify patterns of activity that may be indicative of a security threat, as well as to track the progress of an incident and identify any potential vulnerabilities.

A SIEM system is a software platform that aggregates and analyses data from various sources to identify potential security threats. A Security Orchestration, Automation, and Response (SOAR) system is a tool that automates the response to detected security incidents, including triage, analysis, and resolution. SIEM and SOAR are essential tools within a SOC, as they help streamline and optimise the process of detecting and responding to security threats.

Incident response and investigation involve the steps taken by a SOC to identify, contain, and resolve a security incident. This can include conducting forensic analyses to determine the incident's root cause, communicating with stakeholders and partners, and taking steps to prevent similar incidents.

The technologies and processes used to identify and prevent potential security threats can include firewalls, intrusion detection systems, and other security controls to block or mitigate potential threats. Within a SOC, threat detection and prevention are ongoing activities involving continuous data analysis to identify and respond to potential threats promptly.

Compliance and regulatory requirements

Logging is often an essential aspect of compliance, dictated by regulatory requirements such as the General Data Protection Regulation (GDPR) in Europe and the Protection of Personal Information Act (PoPIA) in South Africa. Logging enables organisations to track and record various types of activity within their systems and networks. This includes user activity, system events, and security-related events.

The GDPR is a European Union (EU) regulation that establishes strict rules for the collection, use, and protection of personal data. Under the GDPR, organisations must maintain detailed records of their data processing activities, including the types of data processed, the purposes for which the data is processed, and the data recipients. PoPIA incorporates similar legislation.

One way logging can help organisations comply with the GDPR is by providing a record of user activity and access to personal data. This can include logs of access to personal data, as well as logs of any modifications or deletions of personal data. By maintaining these logs, organisations can demonstrate that they are properly handling personal data in accordance with the GDPR's requirements.

In addition, the GDPR requires organisations to implement appropriate technical and organisational measures to protect personal data, including measures to detect and prevent unauthorised access or processing of personal data. Logging can play a role in providing a record of security-related events, such as failed login attempts and detected security threats. This can help organisations identify and respond to potential security incidents, as well as demonstrate that they are taking appropriate steps to protect personal data.

Log analysis and monitoring

Log analysis is the practice of looking into event logs produced by computers in order to find vulnerabilities, security concerns, and other dangers early on. Broader applications of log analysis include reviewing user behaviour and ensuring regulatory compliance.

A log is an extensive document that records activity on the device, software program, or operating system. The log file automatically records any data that the system administrators designate, such as sign-in/sign-out requests, error reports, file transfers, requests for files, and messages. In the event of a system failure, security breach, or other anomalous event, the action is further timestamped, which aids developers and IT specialists in creating an audit trail.

Log concepts:

- **Log rotation:** The process of archiving and compressing old log files to manage disk space. Tools like logrotate are commonly used for this purpose.
- **Log analysis:** Tools and techniques used to analyse log data for troubleshooting, monitoring, and security auditing. Examples include Splunk, Graylog, and the ELK stack.
- **Log forwarding:** The process of sending log messages from one system to another. This can be done using Syslog protocols or other log forwarding mechanisms.
- **Centralised logging:** The practice of aggregating logs from multiple sources into a central location for easier management, analysis, and monitoring.

Monitoring

Monitoring is the process of continuously observing apps and systems to make sure they are functioning properly. It aids in keeping things operating at peak efficiency and spotting problems before they become serious.

- **Monitoring performance:** keeping tabs on variables including network traffic, disc input/output, and CPU and memory consumption.
- **Health monitoring:** keeping an eye on how responsive and readily available services and apps are.
- **Monitoring security:** keeping an eye out for any threats, suspicious activity, and unauthorised access.
- **Monitoring compliance:** making sure that systems follow internal guidelines and legal requirements.

Logging levels

Logging levels categorise the importance and type of log messages, helping to filter and prioritise log data based on their severity and significance.

- **DEBUG:** Provides detailed information useful for diagnosing problems or debugging issues.
- **INFO:** Offers general information about system operations and state changes, providing an overview of normal system behaviour.
- **NOTICE:** Indicates normal but significant conditions that don't require immediate action but should be monitored.

- **WARNING:** Highlights potential issues that might need attention to prevent future problems or errors.
- **ERROR:** Indicates error conditions that require immediate attention, such as failures or problems affecting system functionality.
- **CRITICAL:** Represents severe conditions indicating a critical problem that could lead to system or application failure if not addressed promptly.
- **ALERT:** Indicates conditions that require immediate action to prevent further damage or loss.
- **EMERGENCY:** Represents severe conditions that render the system unusable and require urgent intervention to restore functionality.

Understanding these logging levels helps administrators and developers effectively manage and respond to system events, ensuring timely resolution of issues and proactive maintenance of system health.

Log management tools

Software programs known as log management tools are made to gather, handle, and store log data produced by different IT infrastructure devices and applications within an organisation. With the use of these technologies, log data can be tracked, possible problems can be found, and system and device behaviour inside an organisation may be understood. The capacity to search and filter log data, provide reports and alerts, and interface with other IT management tools are common aspects of log management solutions. Some of these tools include:

1. **Splunk:** Splunk is a general-purpose platform designed to handle large volumes of data and can process and index petabytes of data in real time. Offers powerful visualisation, alerting, and reporting capabilities.
2. **ELK Stack:** ELK Stack is a popular open-source log management platform that is composed of three main components: Elasticsearch(A powerful search and analytics engine), Logstash(A log pipeline tool for collecting, processing, and forwarding logs.), and Kibana(A visualisation tool for creating dashboards and reports).
3. **Graylog:** An open-source log management platform that provides a centralised platform for collecting, storing, and analysing log data generated by various devices and applications in an organisation's IT infrastructure. Supports custom alerts, dashboards, and seamless integration with various data sources.

4. **Fluentd**: An open-source data collection solution that provides a logging layer for unified log collection and analysis from many sources. It decouples each data source from the backend system, collecting middleware and application log data and performing log analysis. Supports various input and output plugins for flexible log management.
5. **Prometheus and Grafana**: **Prometheus** is an open-source monitoring and alerting toolkit. **Grafana** is a visualisation tool that integrates with Prometheus for creating dashboards and alerts.

Log management best practices

Efficient and successful log analysis and monitoring are ensured by putting best practices in log management into practice.

1. **Centralised logging**: Logs from several sources should be combined and kept in one place for simple access and examination. Make use of log shippers like Fluentd and Logstash or tools like rsyslog and syslog-ng.
2. **Policies for log retention**: Establish log retention guidelines to control disc space and abide by legal requirements. For automated log rotation and archiving, use software such as logrotate.
3. **Consistent log format**: Make sure that logs from various sources have the same format: For simpler parsing and analysis, use structured logging formats such as JSON.
4. **Enhancement of logs**: To enable more thorough analysis, provide more context to logs, such as user and request IDs and geolocation information.
5. **Security and access control**: Safeguard log data to stop tampering and unwanted access. For log transmission and storage, use encryption.
6. **Real-time tracking and warning**: Configure alerts and real-time monitoring for abnormalities and important occurrences. To send out notifications, use threshold-based and anomaly detection techniques.
7. **Frequent evaluations and audits**: To find trends and enhance monitoring tactics, do routine audits and reviews of log data. To summarise important insights, use dashboards and automated reports.
8. **Documentation and compliance**: Make sure your logging methods adhere to industry norms and legal regulations. Retention guidelines, incident response protocols, and logging setups should all be documented.

Syslog overview

Syslog, an acronym for "System Logging Protocol," is a common protocol for transmitting event or system log messages to a designated server, sometimes called a log server or syslog server. It may be used by a variety of devices and applications, including operating systems, applications, and network devices (routers, switches). It is extensively used for computer system administration and security audits.

How syslog works

Log messages are sent across IP networks in order for syslog to function. For more dependable delivery, it can also utilise TCP, however, it usually uses UDP port 514.

1. **Message generation:** Log messages are produced by devices and programs.
2. **Formatting of messages:** Every message is structured in accordance with the Syslog protocol, which comprises the application name, hostname or IP address of the device, a timestamp, a PRI (priority code), and the message content.
3. **Transmission of messages:** The Syslog server receives the log messages that are transmitted from the clients (devices/applications).
4. **Receiving and processing messages:** The messages are received by the syslog server, which then analyses and stores them in log files or databases for further study and oversight.

Syslog elements

Syslog clients are programs or hardware that produce and transmit log messages. Applications, firewalls, Unix/Linux systems, and routers are a few examples.

The central system that receives, analyses, and stores the log messages supplied by Syslog clients is called the **syslog server**, often referred to as a log server or collector. It may make use of specialised software like rsyslog, syslog-ng, or other log management solutions, and it may run on a variety of operating systems.

- **rsyslog:** An enhanced version of the traditional syslog daemon with additional features like filtering, encryption, and high-performance capabilities.
- **syslog-ng:** A highly configurable logging daemon that supports a wide range of input and output methods, message filtering, and more.
- **Logstash:** Part of the ELK stack (Elasticsearch, Logstash, Kibana), Logstash is a powerful log pipeline tool that can aggregate, process, and forward log messages to various destinations.

Syslog message structure

A typical syslog message has the following structure:

<PRI>timestamp hostname application[pid]: message

- **PRI:** Priority code calculated from the facility and severity levels.
- **timestamp:** The date and time when the log message was generated.
- **hostname:** The name or IP address of the device generating the log.
- **application:** The name of the application or process generating the log.
- **pid:** The process ID of the application generating the log (optional).
- **message:** The actual log message content.

Facilities and severities

Syslog messages include facility and severity levels to categorise and prioritise messages:

Facilities: Indicate the type of source generating the log. Examples include:

- 0: kernel messages
- 1: user-level messages
- 2: mail system
- 3: system daemons
- 4: security/authorisation messages
- 5: messages generated internally by syslogd

Severities: Indicate the importance of the log message. Examples include:

- 0: Emergency – system is unusable
- 1: Alert – action must be taken immediately
- 2: Critical – critical conditions
- 3: Error – error conditions
- 4: Warning – warning conditions
- 5: Notice – normal but significant condition
- 6: Informational – informational messages
- 7: Debug – debug-level messages



Take note

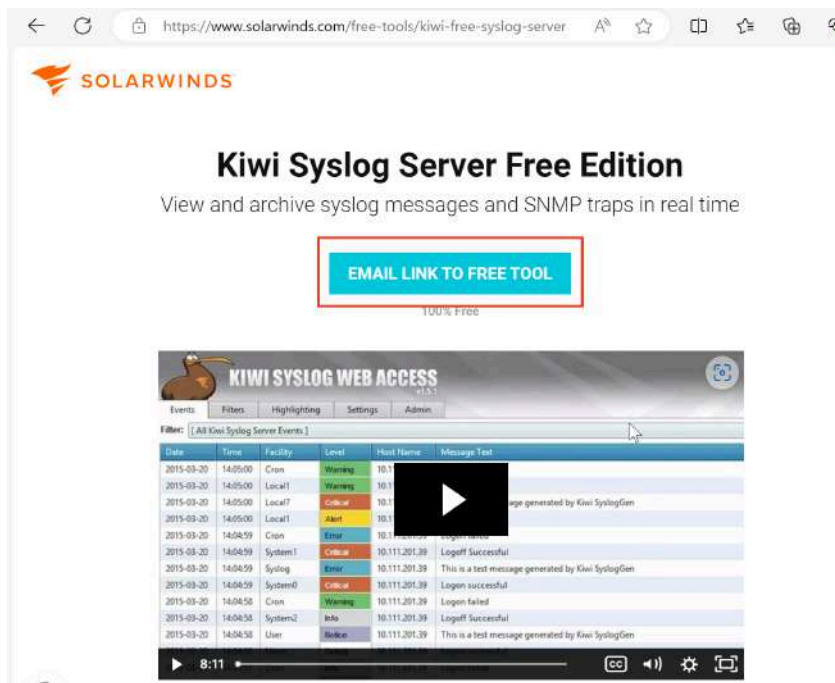
In this practical task, you will install a **syslog server** on your workstation and generate logs to be sent to the server. Three versions of the practical task are provided to cater for Windows, Linux, and macOS users. Please complete the practical task relevant to the operating system on your local machine.



Practical task (Windows)

Follow these steps:

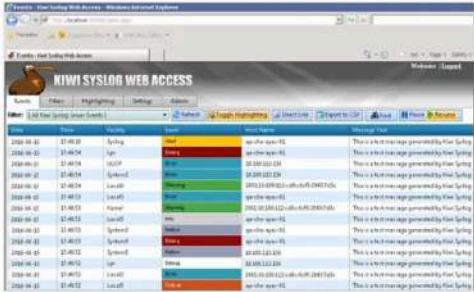
1. Go to the SolarWinds website: [FREE Kiwi Syslog Server Free Edition | SolarWinds](https://www.solarwinds.com/free-tools/kiwi-free-syslog-server).
2. On the website, you will see an option to “Email link to free tool.” Click on this option to proceed to the next page where you will fill in your details, including the email address where you’d like to receive the link.



Once you've filled in your details, you will receive an email with a download link. You can use your personal email and phone number under “Business email” and “Business phone”, and enter your name under “Company name.” Click on the link in the email to download the file.

You will also see the following pop-up options:

Add a Free 14-day Trial of Kiwi Syslog Server NG.




With Kiwi Syslog® Server free edition, you can collect, view, and archive up to five sources, including routers, computers, or other devices.

Have more than five devices to monitor syslog messages and SNMP traps on? Download a FREE 14-day trial of SolarWinds® Kiwi Syslog Server full edition to get unlimited listening.

[ADD KIWI SYSLOG SERVER TO MY DOWNLOAD](#)

[Continue Without Adding](#)

As an alternative, you can select “Continue without adding” as shown above, which will download the tool directly to your computer. Then, click “Download now.”



[Products](#) [Solutions](#) [Resources](#) [Quote](#)

Thank You

We've sent an email with more information to help you get started with your trial install and evaluation. Check your inbox or spam folder.

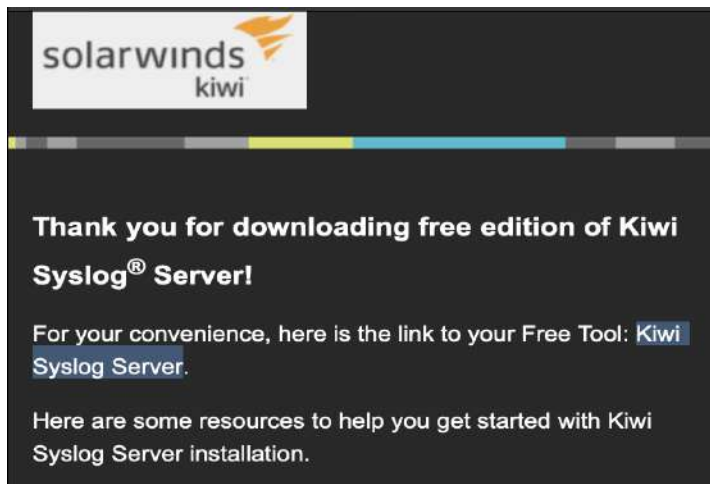
Select Your Download Version

Download Kiwi Syslog Server Free Edition by clicking the button below.

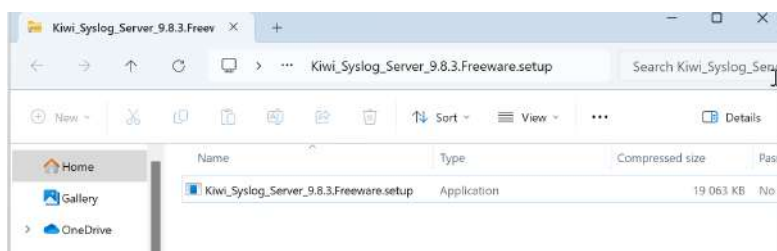
Windows

[DOWNLOAD NOW](#)

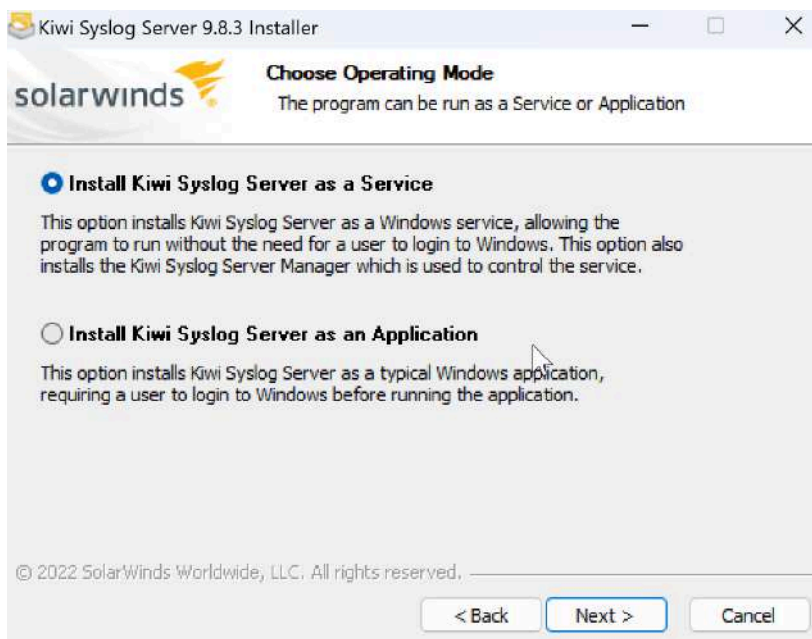
If you choose to close the pop-up, you will still receive the download link in your email. The email will look like this, and you can click the link to download the file.

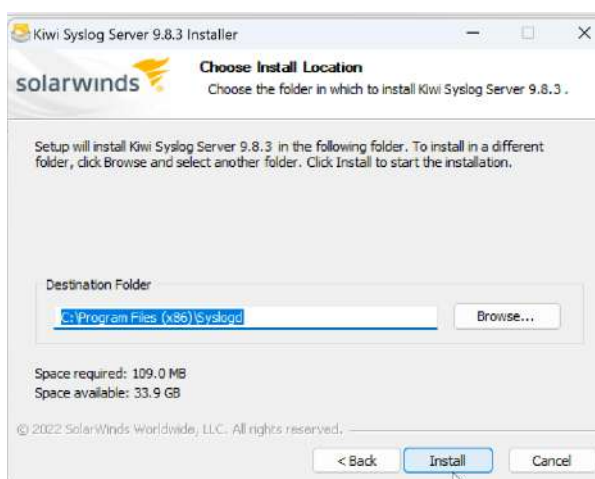
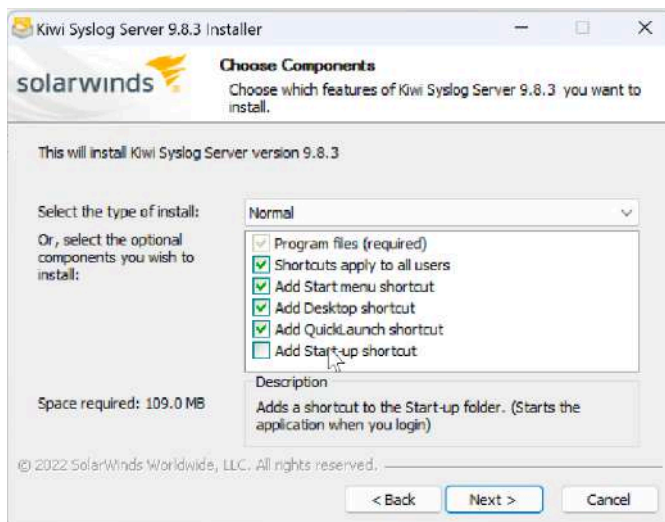


3. Once the download is complete, open the downloaded file and follow the instructions to install Kiwi Syslog Server as an application on your computer. Be sure to accept the terms and conditions when prompted. Here's what it looks like:

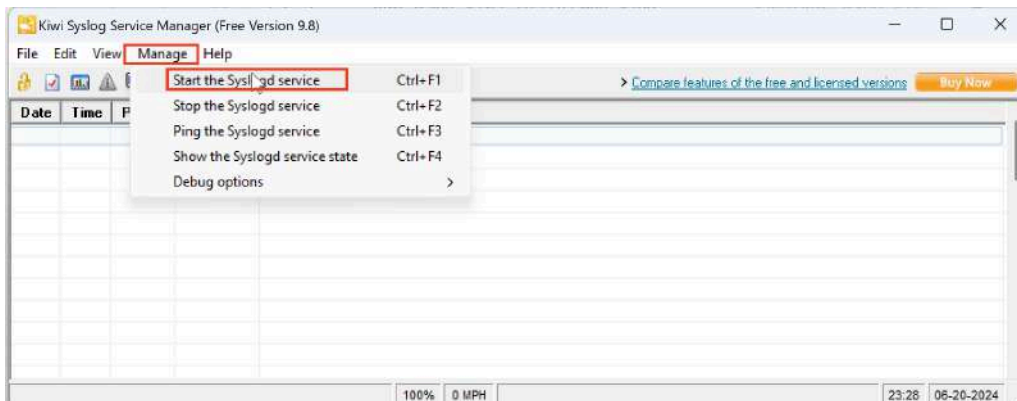


4. Next, begin the installation process as shown below. You can leave the default options unchanged and click "Next" until you reach "Finish."



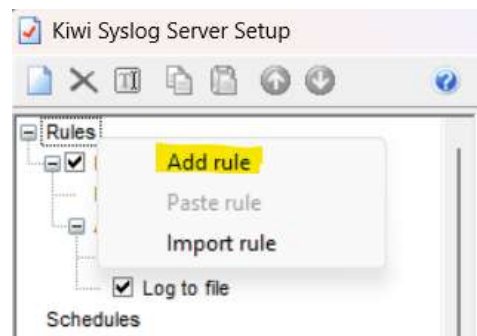
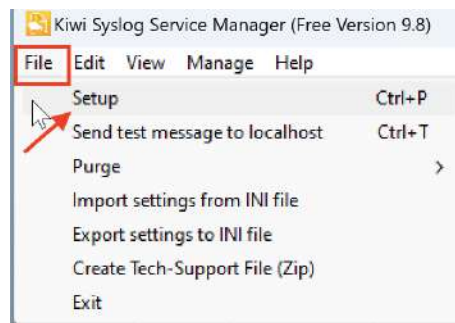


5. Once the installation is complete, you will open the service, click on “Manage” and start the service.

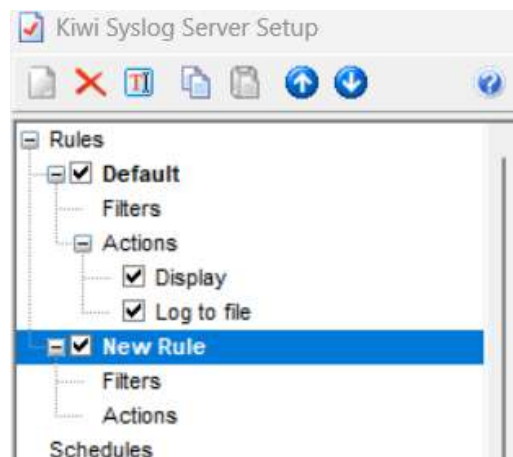


6. Next, you will add a new rule. In the Kiwi Syslog Server application, click on “File” then “Setup”. In the setup pop-up window, right-click on "Rules" and click on the "Add" button to create a new rule for receiving logs.

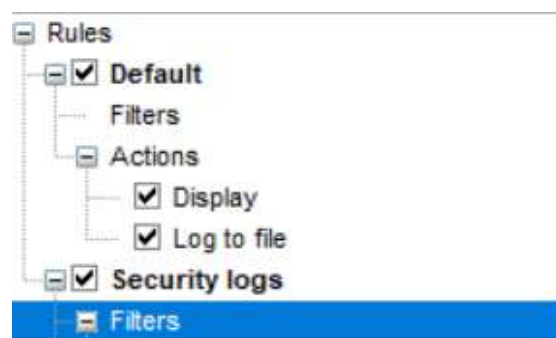
7.



8. A new rule will appear below “Default”. Right-click on “New Rule”, select “Rename”, and rename the new rule “Security Logs”.

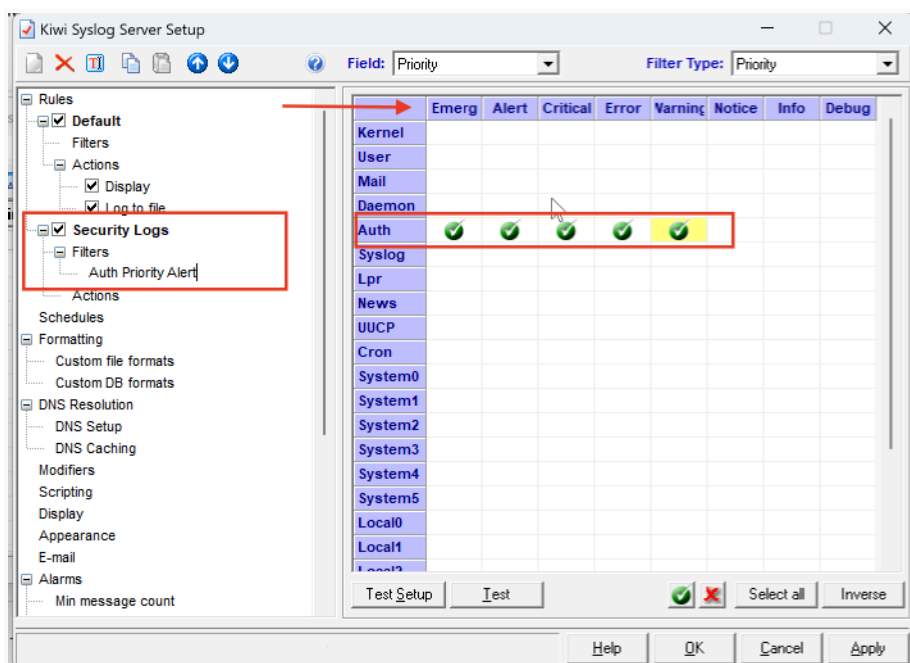


9. Next, you will specify the conditions for matching incoming logs, such as the priority, or time of day. Right-click on “Filters” and add a filter. You can name it “Auth Priority Alert”.



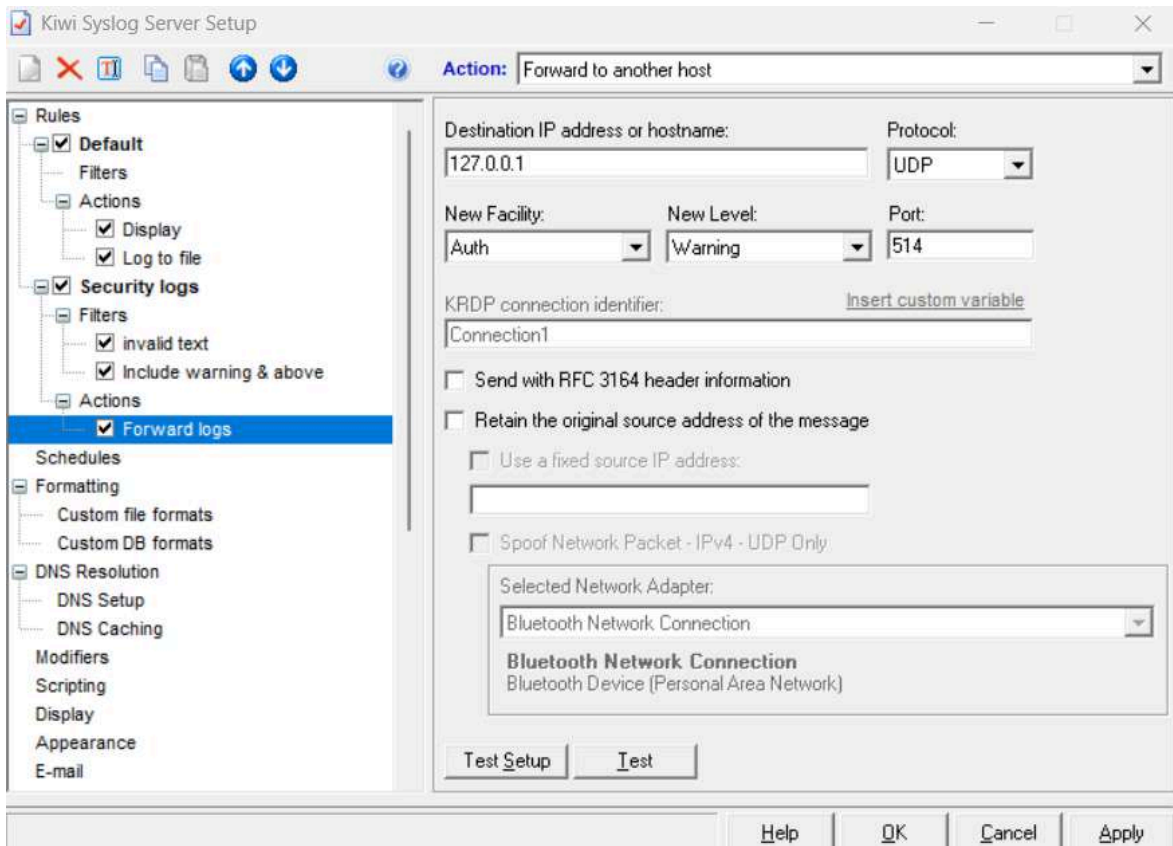
- a. Here are a few examples of conditions in the “Field” dropdown that you can use to match incoming logs:
- i. **Input source:** You can specify the IP address or hostname of the device that is sending the logs. This is only available on the paid version.
 - ii. **Priority:** You can specify the severity level of the log, such as "emergency", "alert", "critical", "error", "warning", "notice", "info", or "debug". These priorities can be selected for the type of system or service that generated the log, such as "auth" for authentication, "cron" for cron jobs, or "local0" for custom applications.
 - iii. **Message text:** You can specify a keyword or phrase that must be present in the log message for the rule to match. You can use **regular expressions** to specify more complex patterns in the “Filter type” dropdown. This is only available on the paid version.

10. Click on this new filter, select the field “Priority” from the dropdown, then scroll down to the “Auth” facility. For the severity level (indicated with an arrow), set the levels to "warning," "error," "critical," "alert," and "emergency." To set these levels, double-click on each one to create green checkmarks, as shown below. Click “Apply” next to apply the filter.



11. Next, add a new action and name it “Forward logs”.

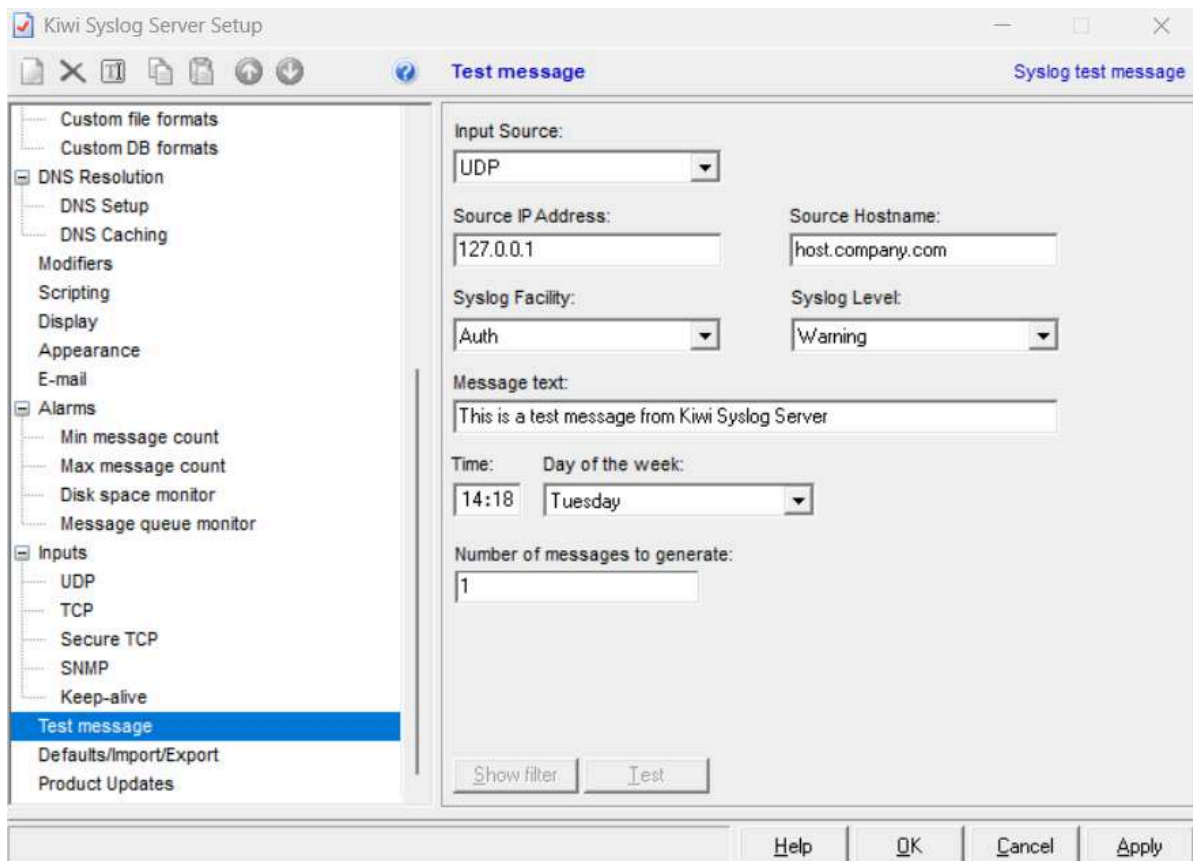
12. In the "Forward logs" action, select the "Forward to another host" option and specify the local machine IP address (127.0.0.1) and port (514) of the syslog server. This allows you to forward security-related logs for further analysis.



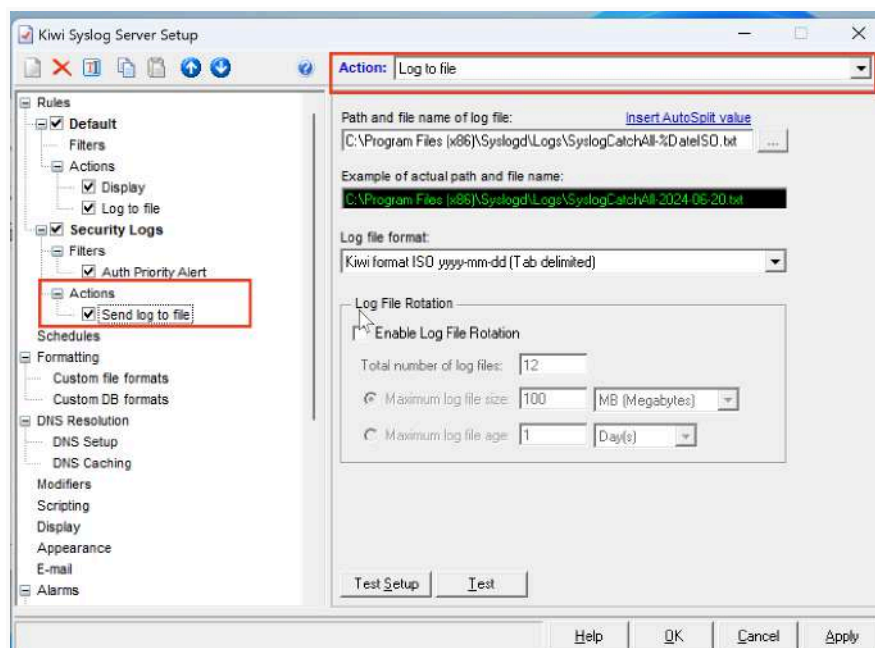
13. Click on the "Apply" button to save the rule and apply it to the local host.
14. You can then view the forwarded logs in the Kiwi Syslog Server application by going to the main window. Note that the "Priority" should be the "Auth" facility as we selected in the previous step.
15. To test your setup, find the test setup section in the left-hand panel and add the same information as in the "Forward logs" action. You can also edit the message text section and add "You have been warned!" to the message. It will show up like the screenshot above when forwarded.

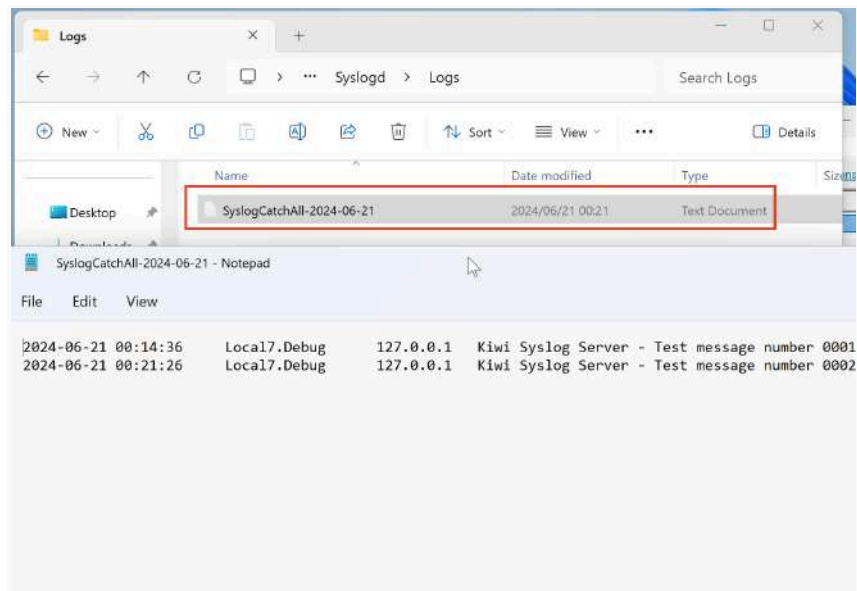
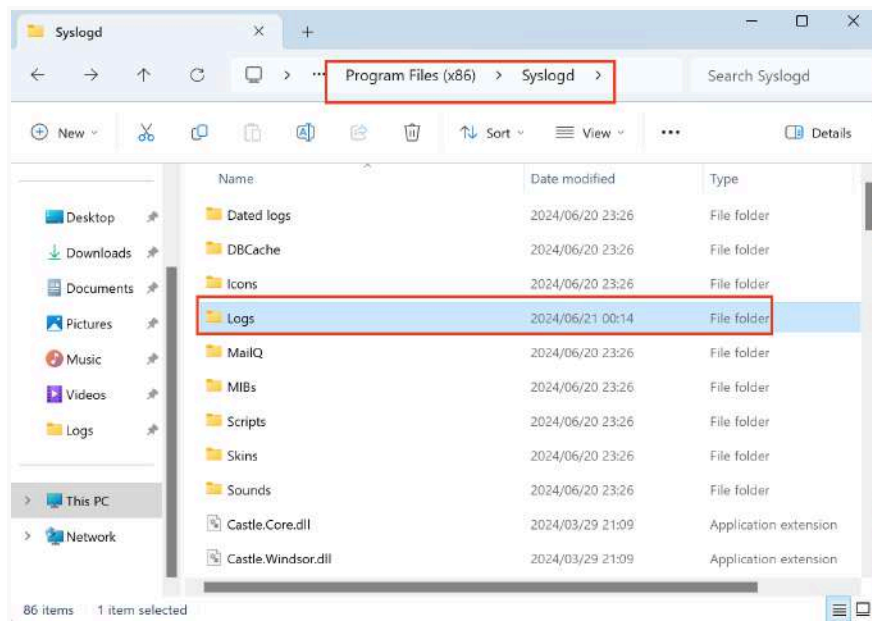
The screenshot shows the 'Kiwi Syslog Service Manager (Free Version 9.8)' window. The menu bar includes File, Edit, View, Manage, and Help. The status bar shows 'Display 00 (Default)' and a 'Buy Now' button. The main area displays a log table with the following data:

Date	Time	Priority	Hostname	Message
06-21-2024	21:30:44	Auth.Warning	127.0.0.1	This is a test message from Kiwi Syslog Server. You have been warned!
06-21-2024	00:21:26	Local7.Debug	127.0.0.1	Kiwi Syslog Server - Test message number 0002
06-21-2024	00:14:36	Local7.Debug	127.0.0.1	Kiwi Syslog Server - Test message number 0001



16. Once you've forwarded the logs to another host, you can also select another action to send to file, create a new action and name it "Send to file", then select "Log to file" on the action which will send the logs results to a file on your computer. Click "Apply" and open your file explorer to see what it looks like.





- Once you have completed this task, please take a screenshot of the main window to show the test message. Save it as a **test_message.jpg** in your task folder. If the test message is being repeatedly logged, go to "Manage" on the taskbar and select "Stop the syslog service".



Practical task (MacOS)

Follow these steps:

1. For macOS, instead of using Kiwi Syslog Server, you can use the built-in **syslogd daemon**. The syslog daemon (syslog) on macOS is configured through the **/etc/syslog.conf** configuration file. Follow the steps below to send all Syslog messages from macOS.
2. Open the Terminal on your Mac.
3. Run the command `sudo nano /etc/syslog.conf` to open the syslog configuration file in the nano text editor. This file contains information used by the system log daemon, syslogd, to forward system messages to appropriate log files and/or users.

```
sh-3.2# nano /etc/syslog.conf
sh-3.2#
```

4. Add the following line at the end of the file: `* . * @127.0.0.1:514`. In this case, the user we are forwarding to is the localhost IP address which would be your machine i.e. 127.0.0.1 (a generic localhost IP) using the UDP port 514. Port 514 is a well-known UDP port for syslog services. Syslog works by sending standardised messages from syslog clients to a syslog server over port 514.

```
UW PICO 5.09      File: /etc/syslog.conf      Modified
# Note that flat file logs are now configured in /etc/asl.conf
install.*                                @127.0.0.1:32376

*. * @127.0.0.1:514
```

It is not necessary to send all syslog messages to the host name. Here's an example of the messages that can be sent to an IP:

An asterisk (*) indicates **all** facilities except for the marked facility. Which is what we used in this case, meaning we are referring to **all logs**.

```
*.emerg    @127.0.0.1:514
*.alert    @127.0.0.1:514
*.crit     @127.0.0.1:514
*.err      @127.0.0.1:514
*.warning  @127.0.0.1:514
```


Examples of facilities:

- **emerg:** For panic conditions that would normally be broadcast to all users.
- **Alert:** For conditions that should be corrected immediately, such as a corrupted system database.
- **Crit:** For warnings about critical conditions, such as hard device errors.
- **Err:** For other errors.
- **Warning:** For warning messages.

5. Press **Control + O** to save the changes and **Control + X** to exit nano.

6. Run the following command to stop the syslog daemon:

```
sudo launchctl stop com.apple.syslogd
```

Then, run the following command to start the syslog daemon with the new configuration:

```
sudo launchctl start com.apple.syslogd
```

```
sh-3.2# launchctl stop com.apple.syslogd
sh-3.2# launchctl start com.apple.syslogd
sh-3.2#
```

This will allow the changes to take effect.

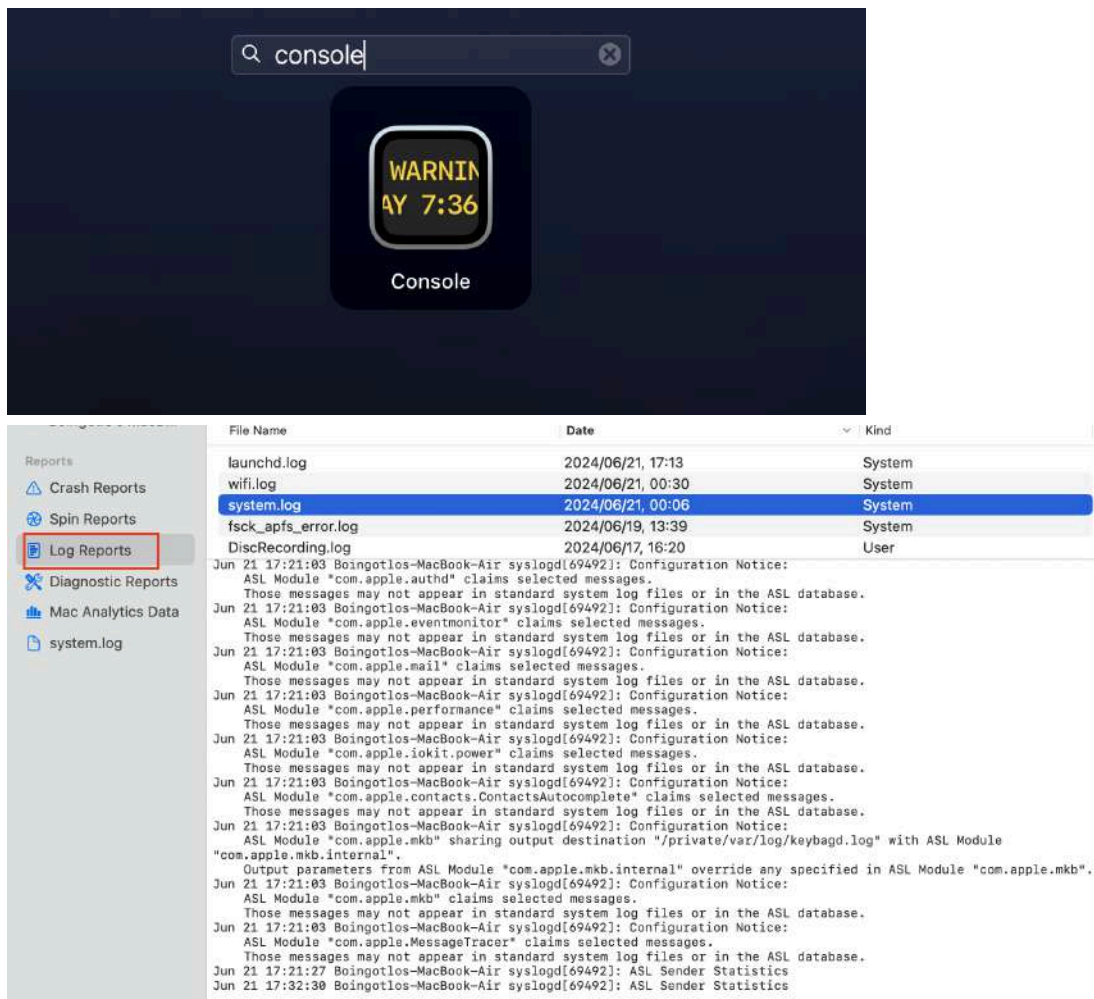
7. Next, run the command `sudo tail -f /var/log/system.log` to view the logs in real time.

The `tail` command allows viewing the last few lines of a file and the `-f` option causes `tail` to not stop when the end of file is reached, but rather to wait for additional data to be appended to the input.

The `var/log/system.log` is where the plain-text log files are stored.

```
sh-3.2# sudo tail -f /var/log/system.log
Jun 21 17:21:03 Boingotlos-MacBook-Air syslogd[69492]: Configuration Notice:
ASL Module "com.apple.mkb" sharing output destination "/private/var/log/keybagd.log" with ASL Module "com.apple.mkb.internal".
Output parameters from ASL Module "com.apple.mkb.internal" override any specified in ASL Module "com.apple.mkb".
Jun 21 17:21:03 Boingotlos-MacBook-Air syslogd[69492]: Configuration Notice:
ASL Module "com.apple.mkb" claims selected messages.
Those messages may not appear in standard system log files or in the ASL database.
Jun 21 17:21:03 Boingotlos-MacBook-Air syslogd[69492]: Configuration Notice:
ASL Module "com.apple.MessageTracer" claims selected messages.
Those messages may not appear in standard system log files or in the ASL database.
Jun 21 17:21:27 Boingotlos-MacBook-Air syslogd[69492]: ASL Sender Statistics
```

You may also find these on your computer by going to “Launchpad” and searching for “Console”. Click on Console and it will bring up different logs. On the side bar look for “Log Reports” and you can view any log from the files. The one shown below is the `system.log` as we did in the terminal, you may also view the other logs for more info. Keep in mind that if you close the terminal, the logs will stop generating here as well.



8. Take a screenshot of the terminal window displaying the logs and save it as **test_message.jpg** in your task folder.

Important: Be sure to upload all files required for the task submission inside your task folder and then click "Request review" on your dashboard.



Practical task (Linux)

For this task, you must have access to two Linux systems. One of them will act as the centralised logging server, while the other will pose as the client that is generating logs. A non-root user with sudo access is required on both systems. We will start with the server.

Rsyslog stands for "Rocket-Fast System for Log Processing," and it is an enhanced version of the traditional syslogd with advanced features such as high performance, modularity, and support for various output formats and protocols

Configuring the Syslog Server

1. Open a terminal and update our system. This is to make sure we are running the current version to avoid any issues

```
sudo apt-get update
```

```
root@bee-ubuntu-linux:/home/bee# apt-get update
Get:1 http://ports.ubuntu.com/ubuntu-ports jammy-security InRelease [129 kB]
Hit:2 http://za.ports.ubuntu.com/ubuntu-ports jammy InRelease
Get:3 http://za.ports.ubuntu.com/ubuntu-ports jammy-updates InRelease [128 kB]
Get:4 http://ports.ubuntu.com/ubuntu-ports jammy-proposed InRelease [279 kB]
Hit:5 http://za.ports.ubuntu.com/ubuntu-ports jammy-backports InRelease
Get:6 http://za.ports.ubuntu.com/ubuntu-ports jammy-updates/main arm64 Packages [1,496 kB]
Get:7 http://ports.ubuntu.com/ubuntu-ports jammy-security/main arm64 Packages [1,297 kB]
Get:8 http://ports.ubuntu.com/ubuntu-ports jammy-security/main Translation-en [261 kB]
Get:9 http://za.ports.ubuntu.com/ubuntu-ports jammy-updates/main Translation-en [319 kB]
Get:10 http://ports.ubuntu.com/ubuntu-ports jammy-security/restricted arm64 Packages [1,463 kB]
Get:11 http://za.ports.ubuntu.com/ubuntu-ports jammy-updates/restricted arm64 Packages [1,511 kB]
Get:12 http://ports.ubuntu.com/ubuntu-ports jammy-security/restricted Translation-en [330 kB]
Get:13 http://ports.ubuntu.com/ubuntu-ports jammy-proposed/main arm64 Packages [224 kB]
Get:14 http://za.ports.ubuntu.com/ubuntu-ports jammy-updates/restricted Translation-en [339 kB]
Get:15 http://ports.ubuntu.com/ubuntu-ports jammy-proposed/main Translation-en [54.0 kB]
Get:16 http://ports.ubuntu.com/ubuntu-ports jammy-proposed/restricted arm64 Packages [333 kB]
Get:17 http://za.ports.ubuntu.com/ubuntu-ports jammy-updates/universe arm64 Packages [1,031 kB]
Get:18 http://ports.ubuntu.com/ubuntu-ports jammy-proposed/restricted Translation-en [71.1 kB]
Get:19 http://za.ports.ubuntu.com/ubuntu-ports jammy-updates/universe Translation-en [251 kB]
Get:20 http://za.ports.ubuntu.com/ubuntu-ports jammy-updates/multiverse arm64 Packages [24.9 kB]
Get:21 http://za.ports.ubuntu.com/ubuntu-ports jammy-updates/multiverse Translation-en [10.8 kB]
Fetched 9,554 kB in 15s (639 kB/s)
Reading package lists... Done
```

2. Once your machine has finished updating, navigate to the rsyslog directory, which contains the **rsyslog.conf** file. Rsyslog can take input from many sources and output it to various destinations. It supports forwarding log messages over an IP network, and the **rsyslog.conf** file allows us to configure these settings. To navigate to the directory, run:

```
cd /etc/
```

Note the rsyslog.conf file is in the directory, along with the rsyslog.d.

```
root@bee-ubuntu-linux:/home/bee# cd /etc
root@bee-ubuntu-linux:/etc# ls
adduser.conf      gai.conf          machine-id         rpc
alsa              gdb               machine-info       rsyslog.conf
alternatives      gdm3              magic              rsyslog.d
anacrontab        geoclue           magic.mime         rygel.conf
```

3. Before editing the **/etc/rsyslog.conf** file, make sure rsyslog is enabled. To do this, ensure you are running as root or use sudo, and enter the following in the terminal:

```
sudo systemctl status rsyslog
```

```
Unknown command verb rsyslog.d.
root@bee-ubuntu-linux:/# systemctl status rsyslog
● rsyslog.service - System Logging Service
   Loaded: loaded (/lib/systemd/system/rsyslog.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2024-06-20 00:22:32 SAST; 49min left
   TriggeredBy: ● syslog.socket
     Docs: man:rsyslogd(8)
           man:rsyslog.conf(5)
           https://www.rsyslog.com/doc/
    Main PID: 701 (rsyslogd)
      Tasks: 5 (limit: 4554)
     Memory: 3.7M
        CPU: 149ms
    CGroup: /system.slice/rsyslog.service
```

If it's not active, run:

```
sudo systemctl start rsyslog
```

4. We need to backup the existing **/etc/rsyslog.conf** file. This is just a backup measure in case a mistake is made. To do this, enter the following in the terminal:

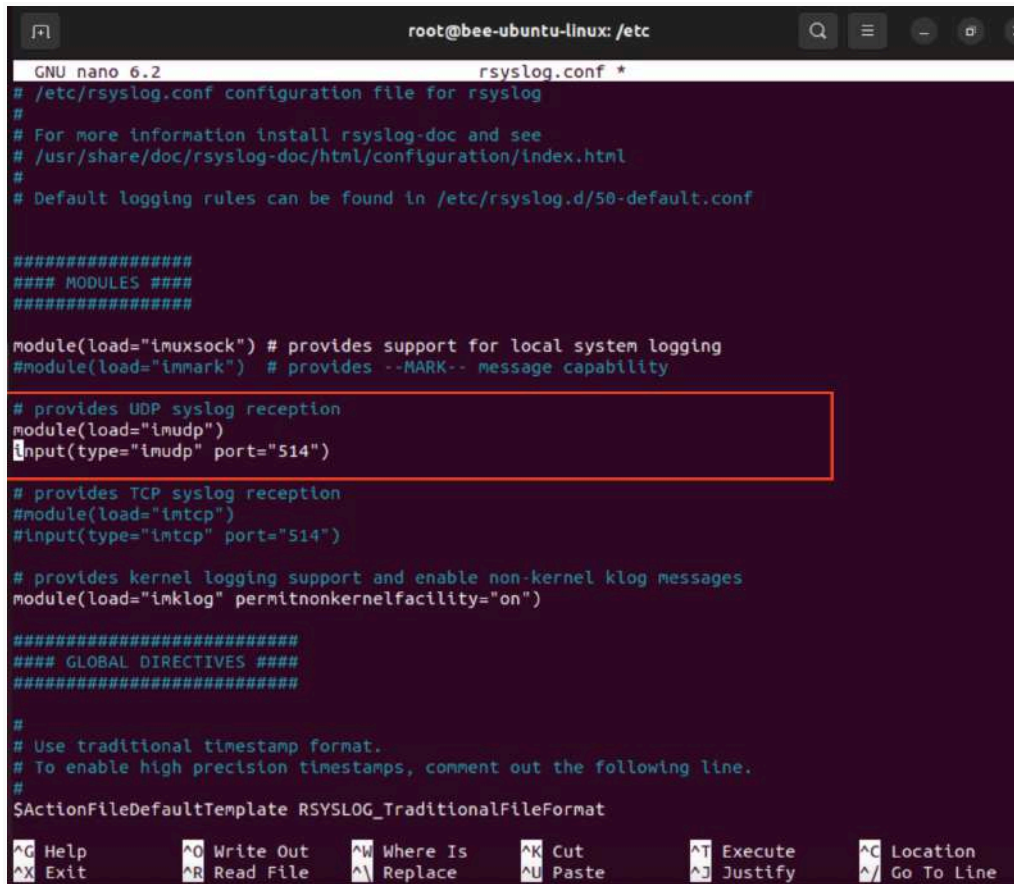
```
sudo cp /etc/rsyslog.conf /etc/rsyslog.orig
```

Then type **ls** to see the copy of the file.

5. Now we will edit the rsyslog.conf file. To do this, enter the following in the terminal:

```
sudo nano /etc/rsyslog.conf
```

Once the file is opened, you will uncomment(remove the #) the UDP section as shown in the screenshot below. This will allow us to receive messages from UDP ports, currently set to port 514 which is a well-known UDP port for syslog services. Exit and save the file after doing this.



```
root@bee-ubuntu-linux: /etc
GNU nano 6.2 rsyslog.conf *
# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index.html
#
# Default logging rules can be found in /etc/rsyslog.d/50-default.conf

#####
#### MODULES ####
#####

module(load="imuxsock") # provides support for local system logging
#module(load="immark") # provides --MARK-- message capability

# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
#module(load="imtcp")
#input(type="imtcp" port="514")

# provides kernel logging support and enable non-kernel klog messages
module(load="imklog" permitnonkernelfacility="on")

#####
#### GLOBAL DIRECTIVES ####
#####

#
# Use traditional timestamp format.
# To enable high precision timestamps, comment out the following line.
#
$ActionFileDefaultTemplate RSYSLOG_TraditionalFileFormat

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^I Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

6. Restart the rsyslog service to initialise your syslog server with the updated config file. This is important in order to load the configurations.

```
sudo systemctl restart rsyslog
```

7. Now we need to check if our server is listening on port 514. We will use the netstat command for this. Netstat is a command-line tool that provides information about network connections, routing tables, interface statistics, masquerade connections, and multicast memberships.

First, install the Netstat tool with this command:

```

sudo apt-get install net-tools
root@bee-ubuntu-linux:/etc# apt-get install net-tools
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 3 not upgraded.
Need to get 207 kB of archives.
After this operation, 774 kB of additional disk space will be used.
Get:1 http://za.ports.ubuntu.com/ubuntu-ports jammy/main arm64 net-tools arm64 1.60+git20181103.0eebece-1ubuntu5 [207 kB]
Fetched 207 kB in 1s (152 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 144718 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-1ubuntu5_arm64.deb ...
Unpacking net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Setting up net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Processing triggers for man-db (2.10.2-1) ...
root@bee-ubuntu-linux:/etc#

```

```
sudo netstat -tuln | grep :514
```

- t: filters TCP ports
- u: filters the output by UDP ports
- l: filters the output by the listening sockets
- n: displays the IP addresses
- **grep :514:** filters the output to show only lines containing:514

```

root@bee-ubuntu-linux:/etc# netstat -tuln |grep :514
udp        0      0 0.0.0.0:514          0.0.0.0:*
udp6       0      0 :::514               :::*
root@bee-ubuntu-linux:/etc#

```

- **udp 0 0 0.0.0.0:514 0.0.0.0:***: Indicates that the server is listening on all IPv4 addresses on UDP port 514
- **udp6 0 0 :::514 :::***: Indicates that the server is listening on all IPv6 addresses on UDP port 514

8. Now for the final part on the server side, we need to configure it to process remote logs by using a template ruleset. By default, all logs that are received from UDP port 514 are usually put in the /var/log directory with the system's log files. This can cause a bit of confusion as it mixes the server's local logs with remote logs which makes it difficult to search and filter when needed. To avoid this, we will edit the rsyslog.conf file again and add the following at the very top of the file as shown below:

```

$template RemoteLogs, "/var/log/%HOSTNAME%/%PROGRAMME%.log"
*.? ?RemoteLogs
& ~

```



```
GNU nano 6.2 rsyslog.conf *
$template RemoteLogs, "/var/log/%HOSTNAME%/%PROGRAMME%.log"
*. * ?RemoteLogs
& ~

# /etc/rsyslog.conf configuration file for rsyslog
#
# For more information install rsyslog-doc and see
# /usr/share/doc/rsyslog-doc/html/configuration/index.html
#
# Default logging rules can be found in /etc/rsyslog.d/50-default.conf

#####
#### MODULES ####
#####

module(load="imuxsock") # provides support for local system logging
#module(load="immark") # provides --MARK-- message capability

# provides UDP syslog reception

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^\ Replace  ^U Paste     ^J Justify  ^_ Go To Line
```

Breakdown of the command:

- **\$template RemoteLogs:** This directive tells rsyslog to store all received log entries in the /var/log directory based on the hostname (client machine name) and the program or application. This means that each client will have a subdirectory inside the /var/log directory with its hostname as the subdirectory's name. Any logs or messages received from this client will be saved here.
- ***. * ?RemoteLogs:** This means the previous rule RemoteLogs will be applied to all (shown by *) facilities (the app/process creating the message) at all severity levels (alert, emerg, critical, etc.)
- **& ~:** This tells the rsyslog to stop processing messages and logs once it is stored to a file defined in previous lines. If this isn't included, then the messages will go to the local files.

9. Save the file and restart the service:

```
sudo systemctl restart rsyslog.service
```

Forwarding Linux Host/client Logs

Now we want to test our server by forwarding logs. To ensure our rsyslog messages are correctly received and processed, we'll set up a host or client to send these messages to our centralised logging server. You'll need a second Linux machine to act as your client. You can use virtual machines for this, such as Ubuntu or Kali Linux, installed through **VirtualBox** or another platform.

1. Before we start, while still in your **server machine**, take note of your server's IP address by typing `ifconfig` in the terminal. Note it down somewhere as you will need it in the following steps.

```

root@bee-ubuntu-linux:/etc# ifconfig
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.32.129 netmask 255.255.255.0 broadcast 192.168.32.255
    inet6 fe80::be8:7d81:a7e6:b3e0 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:fb:1a:96 txqueuelen 1000 (Ethernet)
    RX packets 11772 bytes 16768011 (16.7 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4438 bytes 270936 (270.9 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 44 memory 0x3fe00000-3fe20000

```

2. Now you will login to your other Linux machine and check if the rsyslog service is installed and running with this command:

```
sudo service rsyslog status
```

```

● rsyslog.service - System Logging Service
   Loaded: loaded (/lib/systemd/system/rsyslog.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-06-19 19:25:27 UTC; 19min ago
 TriggeredBy: ● syslog.socket
    Docs: man:rsyslogd(8)
          https://www.rsyslog.com/doc/
   Main PID: 822 (rsyslogd)
     Tasks: 4 (limit: 9516)
    Memory: 7.1M
    CGroup: /system.slice/rsyslog.service
            └─822 /usr/sbin/rsyslogd -n -iNONE

```

If it's not installed (output above not shown), then install it with:

```
sudo apt get rsyslog
```

If it's not running then get it started with this command:

```
sudo systemctl start rsyslog
```

3. We will also navigate to the /etc directory and edit the rsyslog.conf using `sudo nano /etc/rsyslog.conf` in this machine. Inside this file, scroll to the very end and add the following: `*.* @<192.168.32.129:514>`. Take note that this IP address will be different for you based on what you found in step 1 from the server machine. Do not use this specific IP address.
 - The single `@<IP>` indicates that we are using UDP, if we were using TCP it would be `@@<IP>`.
 - `*.*` syntax determines that all log entries on the server should be forwarded. If you want to forward only specific logs, you can specify the service name instead of the asterisk (*) such as `"emerg.* @192.168.32.129:514"` to send only emergency logs. In this step, we are telling the client to forward **all logs** to our server machine (hence the IP address).

```

$WorkDirectory /var/spool/rsyslog
#
# Include all config files in /etc/rsyslog.d/
#
$IncludeConfig /etc/rsyslog.d/*.conf
*.* @192.168.32.129:514

```

4. Restart the service

```
sudo systemctl restart rsyslog
```

5. To test the configuration, we will use the logger command to see if both client and server are configured. The logger command provides an interface into the syslog, which allows us to directly write log entries or messages into the syslog file.

Run `logger TEST TEST TEST` then immediately check if it shows up on the logs by checking the last 10 lines using `sudo tail /var/log/syslog`

```
root@vm-image-ubuntu-dev-1:/home/sysadmin# logger TEST TEST TEST
root@vm-image-ubuntu-dev-1:/home/sysadmin# tail /var/log/syslog
Jun 19 21:14:03 vm-image-ubuntu-dev-1 org.xfce.ScreenSaver[3336]: Xlib: extension "DPMS" missing on display ":10.0".
Jun 19 21:14:09 vm-image-ubuntu-dev-1 root: TEST TEST TEST
Jun 19 21:14:11 vm-image-ubuntu-dev-1 postfix/pickup[5970]: BD6261400E3: uid=0 from=<root>
```

As you can see, our client is able to send logs over to the server using our test log entry, now we should also do the same in the other Linux machine which is the server and we should see the same results.

6. Switch over to your other Linux machine (server) and run the same `logger TEST TEST TEST` command and check the syslog file as above, you should see similar results.

```
Jun 19 22:48:27 bee-ubuntu-linux gnome-shell[1161]: DING: Detected async api for thumbnails
Jun 19 22:48:27 bee-ubuntu-linux gnome-shell[1161]: DING: GNOME nautilus 42.6
Jun 19 22:48:36 bee-ubuntu-linux nautilus[2013]: Could not delete '.meta.isrunning': No such file or directory
Jun 19 22:48:48 bee-ubuntu-linux systemd[1]: forintd.service: Deactivated successfully.
Jun 19 22:48:51 bee-ubuntu-linux bee: TEST TEST TEST
```

This shows that we can see log entries on both the client and server. We have successfully configured the syslog server and forwarded log messages.

7. Lastly, in the same **server** machine run `ls -l /var/log/syslog` to check that the server and client are communicating. You should see a directory that corresponds to the hostname of the **client** system, this directory would have the file of the logs that you just sent over to your server. This confirms that your server is receiving logs from your client.
8. Next, take screenshots of both the server and client machines displaying the log entries to confirm that the logs are being properly forwarded and received. Save these screenshots as **test_message_client.jpg** for the client machine and **test_message_server.jpg** for the server machine. Finally, upload both screenshots to the task folder to complete the verification process.

Important: Be sure to upload all files required for the task submission inside your task folder and then click "Request review" on your dashboard.



Share your thoughts

Please take some time to complete this short feedback [form](#) to help us ensure we provide you with the best possible learning experience.
