Securing MySQL replication on Ubuntu with CIS benchmarks involves several aspects:

1. Secure Communication Channel:

- **Encrypted Connection:** The CIS benchmarks likely recommend establishing an encrypted connection between the MySQL master and replica servers. This protects sensitive data replication traffic from eavesdropping or tampering. Options include:
  - **SSL/TLS:** Configure both MySQL servers to use Secure Sockets Layer (SSL) or Transport Layer Security (TLS) for encrypted communication.
  - **SSH Tunneling:** Set up an SSH tunnel to create a secure channel for replication traffic over an untrusted network.

2. User Accounts and Privileges:

- **Dedicated Replication User:** Create a dedicated user account on the master server specifically for replication purposes. Grant this user only the privileges required for replication, such as REPLICATION SLAVE and minimal other permissions. Avoid using the root user for replication.

3. Secure Configuration:

- **Binary Logging:** Ensure binary logging is enabled on the master server to capture all writeable events required for replication.
- **log_slave_updates:** Enable log_slave_updates on the replica server to record updates received from the master. This can be useful for auditing purposes.
- **read_only:** Set the replica server to read-only mode by default. This prevents accidental writes on the replica that could compromise data consistency.

4. Filtering Replication Traffic (Optional):

- **Replication Filters:** The CIS benchmarks might recommend using replication filters to selectively replicate specific databases, tables, or operations. This can minimize the data transmitted and processing overhead on the replica server.

5. Security Considerations for Replication Slaves:

- **Network Access:** Restrict network access to the replica server to minimize its attack surface. Ideally, only the master server and authorized monitoring tools should be able to connect to the replica.
- **Backups:** Regularly back up the replica server to ensure data recovery in case of incidents.

Here's an example configuration on the master server (replace 'replication_user' and 'strong_password' with your actual credentials):

SQL

```
CREATE USER 'replication_user'@'master_server_ip' IDENTIFIED BY
'strong_password' REQUIRE NONE;
GRANT REPLICATION SLAVE ON *.* TO
'replication_user'@'master_server_ip';
FLUSH PRIVILEGES;
```

Additional Resources:

- CIS MySQL Security Benchmarks (reference security guides that mention these benchmarks): While not directly available online, security guides referencing CIS benchmarks can offer details. Search for "CIS MySQL Security Benchmarks Ubuntu"
- MySQL documentation on replication: https://dev.mysql.com/doc/refman/8.0/en/replication.html

Remember:

- Consult the specific CIS benchmarks for your Ubuntu version for detailed recommendations on replication security.
- Regularly review and update your replication configuration to maintain a secure environment.
- Consider additional security measures like access controls and monitoring for the replica server.