To set up a basic archive server on Ubuntu, we can use Python's built-in http.server module. This approach is simple, scalable, and modular. Let's break it down into steps:

Create the main server script:
archive_server.py

```
import http.server
import socketserver
import os

# Set the directory to serve files from
DIRECTORY = "/home/sauron/quantum_archive"

# Change the working directory
os.chdir(DIRECTORY)

# Set the port for the server
PORT = 8000

# Create handler for serving files
Handler = http.server.SimpleHTTPRequestHandler

# Create the server
with socketserver.TCPServer(("", PORT), Handler) as httpd:
    print(f"Serving files from {DIRECTORY} on port {PORT}")
    httpd.serve_forever()
```

This script sets up a basic HTTP server that serves files from the specified directory.

Create a configuration module:
config.py

```
# Server configuration
SERVER_PORT = 8000
ARCHIVE_DIR = "/home/sauron/quantum_archive"

# Logging configuration
LOG_FILE = "archive_server.log"
LOG_LEVEL = "INFO"
```

Create a logging module:
logger.py

```
import logging
from config import LOG_FILE, LOG_LEVEL
```

```
def setup_logger():
    logging.basicConfig(
        filename=LOG_FILE,
        level=getattr(logging, LOG_LEVEL),
        format='%(asctime)s - %(levelname)s - %(message)s'
    )
    return logging.getLogger(__name__)

logger = setup_logger()
```
Update the main server script to use the config and logger:
```
# archive_server.py

import http.server
import socketserver
import os
from config import SERVER_PORT, ARCHIVE_DIR
from logger import logger

os.chdir(ARCHIVE_DIR)

Handler = http.server.SimpleHTTPRequestHandler

with socketserver.TCPServer(("", SERVER_PORT), Handler) as httpd:
    logger.info(f"Serving files from {ARCHIVE_DIR} on port {SERVER_PORT}")
    httpd.serve_forever()
```
To run the server, navigate to the directory containing these files in the terminal and run:

```
python3 archive_server.py
```

This setup provides a basic, modular foundation for your archive server. You can easily extend it by adding more modules for specific functionalities as your project grows.