

بخش اول

کسرا نوربخش 810100230

مقدمه

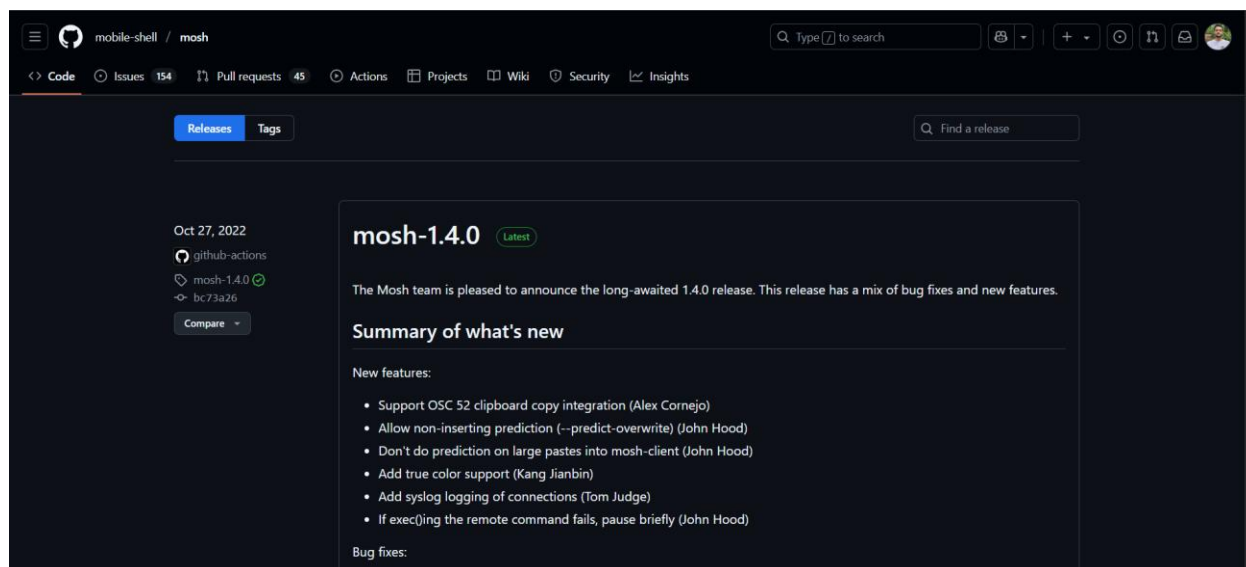
در این فایل به جواب بخش اول یعنی اجرای mosh بدون دسترسی root در سرور پرداختیم.

پاسخ

برای استفاده کردن از mosh به جای ssh در سناریو ای که نیاز داریم remote access به سروری داشته باشیم، به یک mosh-client (که ارتباط را از سر می گیرد و بر روی دیوایس local ما باید نصب باشد) و یک mosh-server (که بر روی سرور اجرا می شود و ارتباط و state آن را نگه می دارد) نیاز داریم.

حال که ما mosh-server را نصب نداریم 2 کار می توان کرد. یک اینکه از ادمین سرور بخواهیم که با استفاده از package manager ها و یا هر روش دیگری که می داند، mosh-server را برای ما نصب بکند تا دیگر مشکلی نداشته باشیم. راه حل دوم که مناسب تر برای این سوال می باشد آن است که ما خودمان پکیج یا همان سورس mosh را از سایتشان دانلود کرده (از طریق این لینک: <https://github.com/mobile-shell/mosh/releases/download/mosh-1.4.0/mosh-1.4.0.tar.gz> که از خود صفحه گیت هاب آن برداشتم)، و آن را در home dir خودمان build و compile کرده و اجرا کنیم. البته این مستلزم آن است که سرور یک کامپایلر و dependency های لازم را داشته باشد. البته این هم با فرض آنکه اجازه اجرای فایل های اجرایی را از home dir ها در سرور داشته باشیم.

پس به ترتیب، دانلود سورس کد mosh از طریق صفحه گیت هابشان (<https://github.com/mobile-shell/mosh/releases/download/mosh-1.4.0/mosh-1.4.0.tar.gz>)، خارج کردن از حالت فشرده و build کردن سورس کد با استفاده از دستور های: `tar -xvzf mosh-1.4.0.tar.gz && cd mosh-1.4.0 && ./configure --prefix=$HOME/mosh && make && make install`، و در نهایت اجرا در home dir مراحل مورد نیاز ما، خواهند بود.



عکس 1: صفحه گیت هاب mosh

لینک های استفاده شده در این بخش

1. <https://gist.github.com/lazywei/12bc1669dc7739dccef1>
2. https://grok.com/share/bGVnYWN5_725b00a2-eb94-4764-9819-66e0781fc9e8

استفاده از هوش مصنوعی Grok 3 با حالت DeepSearch

سوال این بخش را از آن به صورت:

We want to have a connection to a server using mosh. The problem here is that mosh-server app is not installed on the server and we don't have root access. how can we use mosh to connect?

پرسیدم که چت کامل در لینک 2، در بالا موجود است.

چت کامل:

Key Points

- It seems likely that you cannot use mosh to connect to a server without the mosh-server installed, especially without root access, as the server component is essential.
- Research suggests there are workarounds, like compiling mosh-server from source in your home directory, but success depends on server permissions and configuration.

The evidence leans toward needing cooperation from the server administrator if •
mosh-server cannot be installed or run.

Understanding the Challenge

Mosh (mobile shell) is a remote terminal application designed for better responsiveness and reliability compared to SSH, especially over unstable networks. It uses a client-server model, where the mosh-client runs on your local machine, and the mosh-server must run on the remote server you're connecting to. The issue here is that the mosh-server app isn't installed on the server, and you lack root access, meaning you can't install it system-wide.

Exploring Workarounds

Given the constraints, traditional installation via package managers like apt or yum isn't possible without root access. However, there are potential workarounds:

Compiling from Source: You might compile the mosh-server from source and run it •
from your user home directory, avoiding system-wide installation. This requires the server to have a compatible compiler (e.g., gcc) and necessary dependencies, which may not always be available without root.

Checking Server Permissions: Some servers allow users to run executables from •
their home directories, but this depends on the system's security policies and whether the server administrator permits such operations.

Administrator Assistance: If neither of the above is feasible, you may need to •
request the server administrator to install mosh-server, as they have the necessary privileges.

Unexpected Detail: Limited Control Without Root

An unexpected aspect is how limited your control is without root access, highlighting the importance of server policies and administrator cooperation in such scenarios. This can be frustrating, especially for users accustomed to more flexible environments.

Survey Note: Detailed Analysis of Using Mosh Without Root Access

This section provides a comprehensive exploration of the challenges and potential solutions for using mosh to connect to a server where the mosh-server is not installed and

root access is unavailable. The analysis is grounded in the understanding of mosh's architecture and the implications of lacking administrative privileges.

Background on Mosh

Mosh, or mobile shell, is a remote terminal application that enhances the user experience over SSH, particularly in scenarios with high latency or intermittent connectivity. It operates on a client-server model:

The **mosh-client** runs on the local machine, initiating the connection. •

The **mosh-server** must be running on the remote server to handle the connection and maintain state. •

For mosh to function, both components must be present and compatible. The server component, mosh-server, typically needs to be installed via system package managers (e.g., apt install mosh on Debian-based systems) or compiled from source. However, the absence of mosh-server on the target server, combined with the lack of root access, poses a significant barrier.

Challenges Without Root Access

Root access is typically required to install system-wide software, as it involves writing to system directories (e.g., /usr/bin or /usr/local/bin) and managing dependencies. Without it:

You cannot use standard package managers to install mosh-server. •

System security policies may prevent running executables from user directories, depending on the server configuration. •

Necessary build tools (e.g., gcc, make) and libraries may not be available or accessible without root, complicating compilation from source. •

Potential Workarounds

Given these constraints, several strategies can be considered, though their feasibility varies:

Compiling Mosh-Server from Source in User Space .1

It is possible to download the mosh source code from the official GitHub repository ([Mosh GitHub](#)) and compile it within your home directory. This avoids system-wide installation but requires: ○

A compatible compiler (e.g., gcc) and build tools, which may not be available without root. ▪

All necessary dependencies (e.g., protobuf, ncurses) to be either pre-
installed or manually compiled, which can be complex. ■

Permission to run the compiled binary, which depends on server
policies. ■

Example steps include: ○

Download the source: `wget https://github.com/mobile-
shell/mosh/releases/download/mosh-1.4.0/mosh-1.4.0.tar.gz` ■

Extract and compile: `tar -xzf mosh-1.4.0.tar.gz && cd mosh-1.4.0 &&
./configure --prefix=$HOME/mosh && make && make install` ■

Run the server from your home directory, e.g.,
`$HOME/mosh/bin/mosh-server`. ■

Success depends on the server environment; some systems restrict
executable permissions, and missing dependencies can halt the process. ○

Requesting Administrator Installation .2

If compiling is not feasible, the most reliable solution is to request the server
administrator to install mosh-server. This ensures compatibility and proper
system integration, avoiding potential security or permission issues. ○

Administrators may be willing to install it if there is a legitimate need,
especially for shared hosting environments. Communication is key, and
explaining the benefits (e.g., better connectivity for remote work) may help. ○

Alternative: Using SSH with Screen or TMUX .3

If mosh cannot be used, consider alternatives like SSH combined with
session persistence tools like screen or tmux. These allow you to detach and
reattach sessions, mitigating some of mosh's advantages but providing a
fallback: ○

Install screen or tmux if available (may still require root, depending on
the server). ■

Start a session: `tmux new -s my session`, detach with `Ctrl-b d`, and
reattach later with `tmux attach -t my session`. ■

This doesn't replicate mosh's roaming capabilities but can be a practical
workaround for maintaining sessions over unstable connections. ○

Feasibility Assessment

The feasibility of these workarounds depends on several factors, summarized in the following table:

Approach	Requirements	Feasibility Without Root	Notes
Compile from Source	Compiler, dependencies, executable permissions	Low to Medium	Success depends on server tools and policies; may fail due to restrictions.
Request Administrator Install	Administrator cooperation	High (if approved)	Most reliable but depends on external party; may take time.
Use SSH with Screen/TMUX	Screen/TMUX installed, SSH access	Medium (if tools available)	Does not fully replace mosh but mitigates some issues; may need root for install.

Unexpected Detail: Server Policy Impact

An interesting aspect is how server policies can significantly impact user flexibility. For instance, some servers disable user-compiled binaries for security, while others allow it, creating variability in what's possible. This underscores the importance of understanding the specific server environment and potentially negotiating with administrators.

Conclusion

In summary, using mosh without the mosh-server installed and without root access is challenging but not impossible. Compiling from source in your home directory is a potential solution, though its success depends on server permissions and available tools. The most reliable approach is likely requesting administrator assistance, ensuring proper installation and compatibility. If mosh remains unfeasible, alternatives like SSH with screen or tmux can provide partial functionality. Given the current time (02:49 AM PDT on Wednesday, March 26, 2025), consider reaching out to administrators during business hours for assistance.

Key Citations

[Mosh GitHub release page with source code](#) •