Role Based Access Control in SSH

552 admin March 21, 2025 Features & Functionalities 2929

Role-Based Access Control (RBAC) and it's implementation in SSH

Overview: This article explains implementing Role-Based Access Control (RBAC) in SSH using Ezeelogin to restrict user access, manage permissions, and enhance security through an SSH gateway with MFA integration.

Role-Based access control (RBAC) is a security model which is used to manage data and user access in organizations; providing control over the access permissions and simplifying the management to comply with the requirements.

How Role-based access control can be implemented in SSH?

Implementing Role-Based Access Control (RBAC) in SSH ensures that SSH users or administrators have only the minimal access required to perform their duties on the server. In other words, role-based access control (RBAC) in SSH is a method used to restrict SSH users' access or server administrators' access to remote servers based on their specific roles. Most enterprises manage thousands of servers, making granting SSH access to employees a significant challenge and security concern. By using role-based access access control (RBAC), organizations can ensure that SSH users or system administrators only access the relevant information necessary to perform their tasks. Additionally, organizations can restrict access to specific groups of servers and control the actions users can take on those servers.

- **Key Principles** include *role based,* permission based, separation of duties, the centralized, auditable trial of users, and dynamic changes.
- **Key Components** include Users, Permissions, and Roles. It allows flexible management of user access by allowing and modifying roles and permissions as per need.

Benefits of Role-based access control in SSH

When we talk about the benefits of Role based access control (RBAC), being a powerful security model it helps in improved security, Flexibility, Scalability, Cost-effective, and Simplified Access management.

SSH Gateway and Its Connection with Role-Based Access Control (RBAC)

An **SSH Gateway** plays a crucial role in securing SSH access within a network. *SSH gateway* uses a secure shell protocol to securely connect to other servers within a network; acting as a mediator between a user's computer and a target server facilitating secure remote connections.

Integrating **Role Based Access Control** with an SSH gateway further strengthens security by controlling what each user can do once they are inside the network. Instead of allowing unrestricted access, role based access control can limit a user's actions even once they authenticate through the SSH gateway. For example, a user with limited permissions may be able to connect to the gateway but can only access a specific set of servers or perform certain actions on them, based on their role.

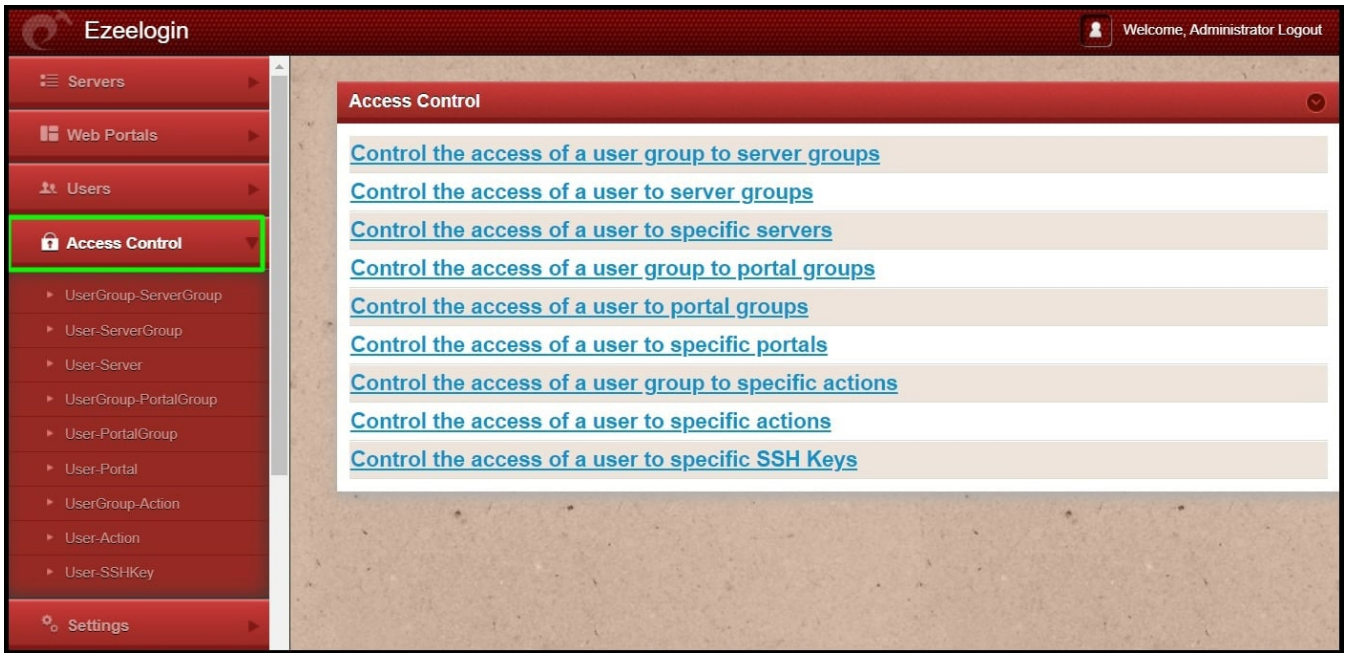
Importance of Implementing proper security measures in SSH Gateway

Key reasons for taking security measures in ssh gateway would be to prevent data breaches, mitigate the risk of attacks, preventing unauthorized access which reduces overall risks of threats.

How to configure role based access control?

To achieve this, we can group the servers and SSH users into different categories based on our requirements and we can decide which user or user group can get access to which server group. Ezeelogin has the feature RBAC. Using this, you can configure role based access control in ssh.

Configuring rbac **in SSH using ezeelogin jump server**



Different ways to achieve Role based access control (RBAC) in ssh :

- Restrict user actions on the server
- Map ssh user to a particular server
- Map ssh user to a group of servers
- Map ssh user group to a single server
- Map ssh user group to group of servers

Adding and removing groups in RBAC.

Use the drop-down menu to choose the Group whose accessibility needs to be altered. Then remove or Add accordingly.

Limiting the number of users with administrative privileges.

Admin user can provide permissions to limit the number of user access on certain servers and groups as per need.

You can also implement multi-factor authentication (MFA) in users for added security

Refer detailed article for user authentication - MFA authentication for user

---

Final thoughts and recommendations.

Regularly reviewing and updating roles and permissions.

Limits unnecessary user access to any servers.

Regulate access of user to SSH keys.

Related Articles:

Tutorial on Role Based Access Control

User manual