User Authentication Protocol in SSH

Message Formats, Exchange and Real-World Case Study

GROUP NO 5

Department of Computer Science

February 20, 2026

Outline

1. Introduction
2. User Authentication Protocol
3. Message Types and Formats
4. Message Exchange
5. Authentication Methods
6. Case Study: SSH in Cloud Systems
7. Conclusion
8. References

Secure Shell (SSH)

- SSH provides secure remote login and communication
- Protects data over insecure networks
- Widely used in servers and cloud platforms

Security Services:

- Confidentiality
- Integrity
- Authentication

What is User Authentication?

- Verifies identity of the client
- Prevents unauthorized access
- Mandatory before accessing SSH services

User Authentication Protocol

- Operates above SSH Transport Layer
- Authenticates client to server
- Defined in RFC 4252
- Supports multiple authentication methods

Purpose of the Protocol

- Validate user identity
- Enable multi-factor authentication
- Secure access to SSH services

Authentication Request (Client)

SSH_MSG_USERAUTH_REQUEST (50)

```
byte    SSH_MSG_USERAUTH_REQUEST
string  user name
string  service name
string  method name
...     method-specific fields
```

- User name: Claimed identity
- Service name: Requested service
- Method name: Authentication method

Authentication Failure (Server)

SSH_MSG_USERAUTH_FAILURE (51)

```
byte      SSH_MSG_USERAUTH_FAILURE
name-list authentication methods
boolean   partial success
```

- Lists supported authentication methods
- Partial success enables multi-step authentication

Authentication Success (Server)

SSH_MSG_USERAUTH_SUCCESS (52)

- Sent when all authentication steps succeed
- Marks end of authentication phase

Authentication Message Exchange

1. Client sends request with method *none*
2. Server validates username
3. Server returns allowed authentication methods
4. Client selects and performs authentication
5. Server verifies credentials
6. Server sends success message

Public Key Authentication

- Client sends public key and digital signature
- Signature generated using private key
- Server verifies key and signature
- Most secure and widely used method

Password Authentication

- Client sends plaintext password
- Encrypted by SSH Transport Layer
- Simple but less secure
- Often disabled in production systems

Host-Based Authentication

- Authentication based on client host
- Host signs request using private key
- Server trusts client host
- Used in controlled environments

Case Study Overview

Scenario

Cloud providers manage thousands of servers requiring secure remote access.

- Large-scale Linux infrastructure
- Multiple administrators
- High security requirements

Implementation

- Public key authentication enforced
- Password authentication disabled
- Multi-factor authentication enabled
- Unique key pairs per administrator

Authentication Workflow

1. Administrator initiates SSH connection
2. Server requests authentication
3. Client sends signed public key
4. Server verifies key
5. Secure access granted

Benefits

- Strong security
- Protection against brute-force attacks
- Scalable user management
- Secure remote administration

Conclusion

- SSH User Authentication ensures secure access
- Supports multiple authentication methods
- Essential in cloud and enterprise systems
- Reliable and widely adopted protocol

References

- RFC 4252 – SSH Authentication Protocol
- William Stallings, Cryptography and Network Security
- OpenSSH Documentation

Thank You