LevenTransfer

File Transfer Protocol

Ankit Sanghi, Mark Rubakh, Winston Wang,
Aurnov Chattopadhyay, Snow Kang

1/ Design Overview

- Attacker Models
- Security Requirements

Attacker Models and Trust Assumptions

Attacker Capabilities

[1] Eavesdropping

[2] Modification and Interception of Messages

[3] Sending Original Messages

Motivations

[1] Impersonation

[2] Identifying File System Characteristics

[3] Jamming

Trust Assumptions

[1] Out of Band Public-Private Key Pair

[2] Pre-registration of Users with Strong Passwords

[3] Cryptographic Primitive Validity

Our Protocol Offers

1 Confidentiality

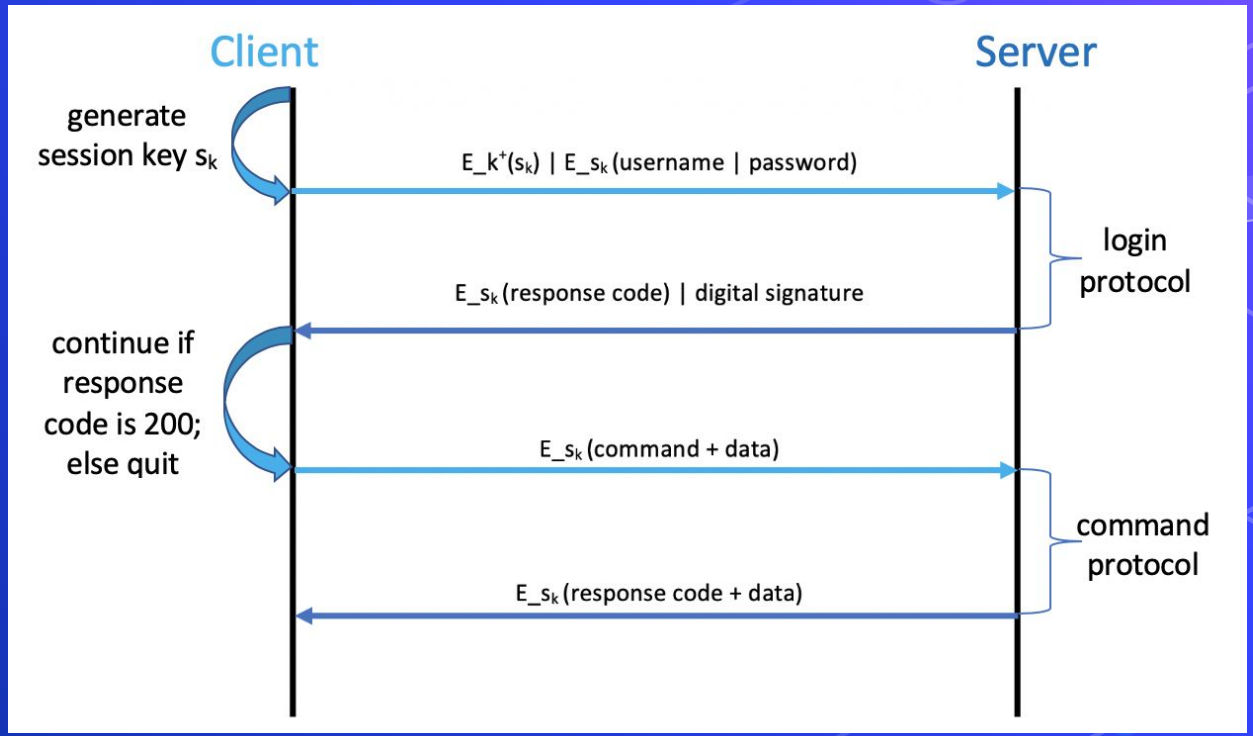
Integrity Protection & Authentication 2

3 Replay Protection

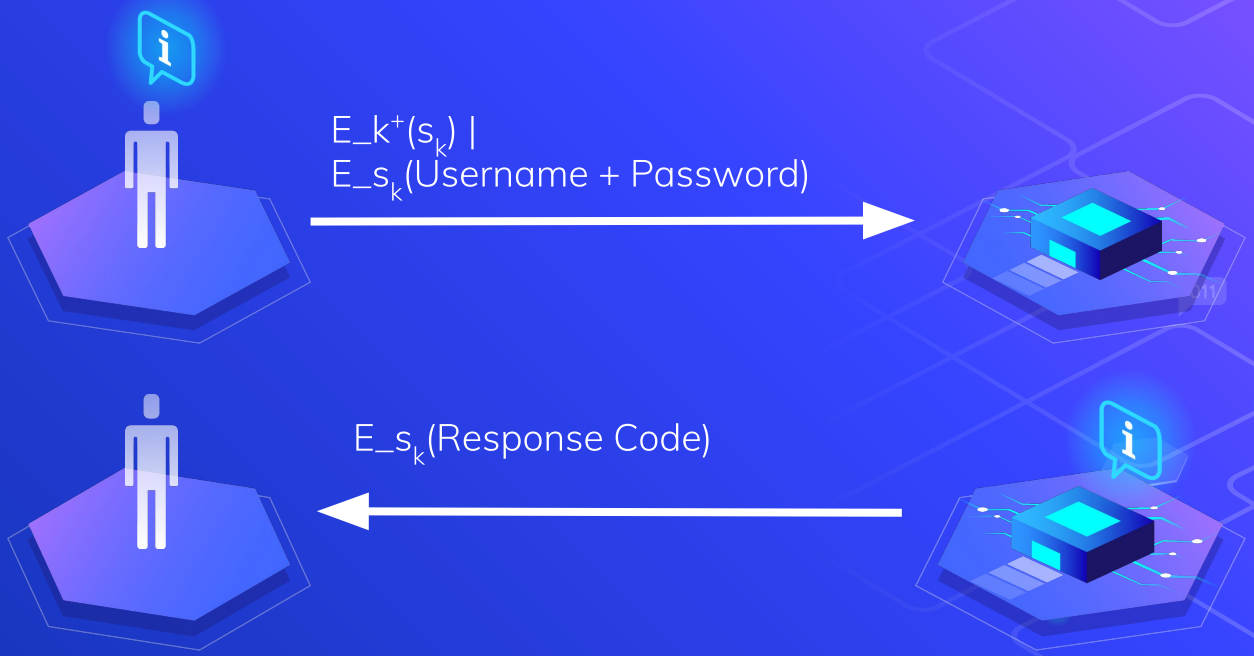
2/ Protocol Overview

- Login Protocol
- Command Protocol
- System Design

Protocol Overview

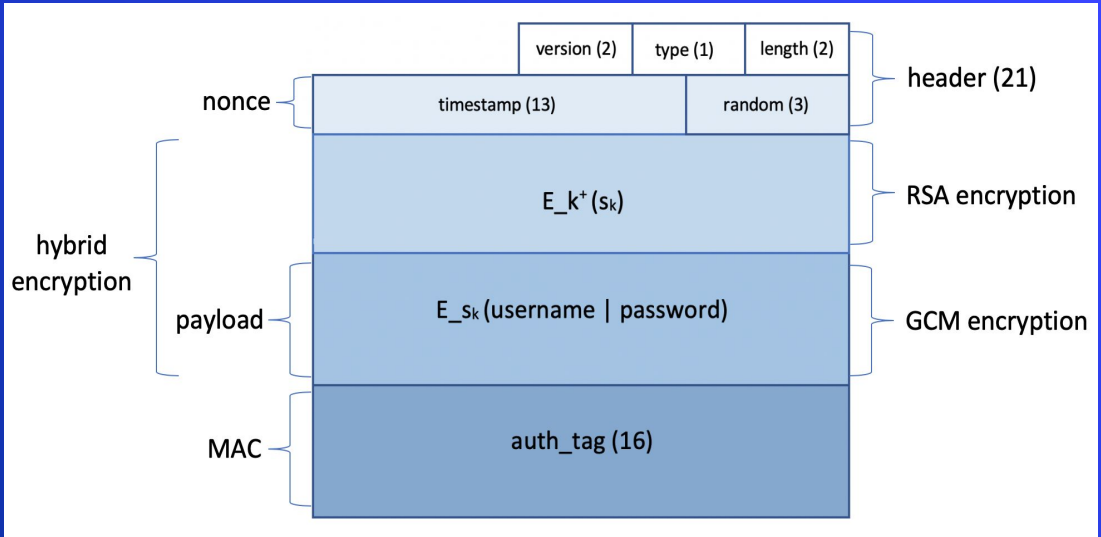


Login Protocol

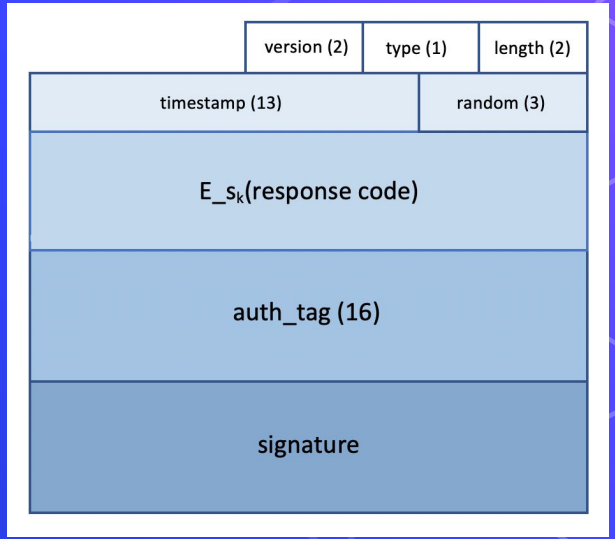


Login Protocol Format

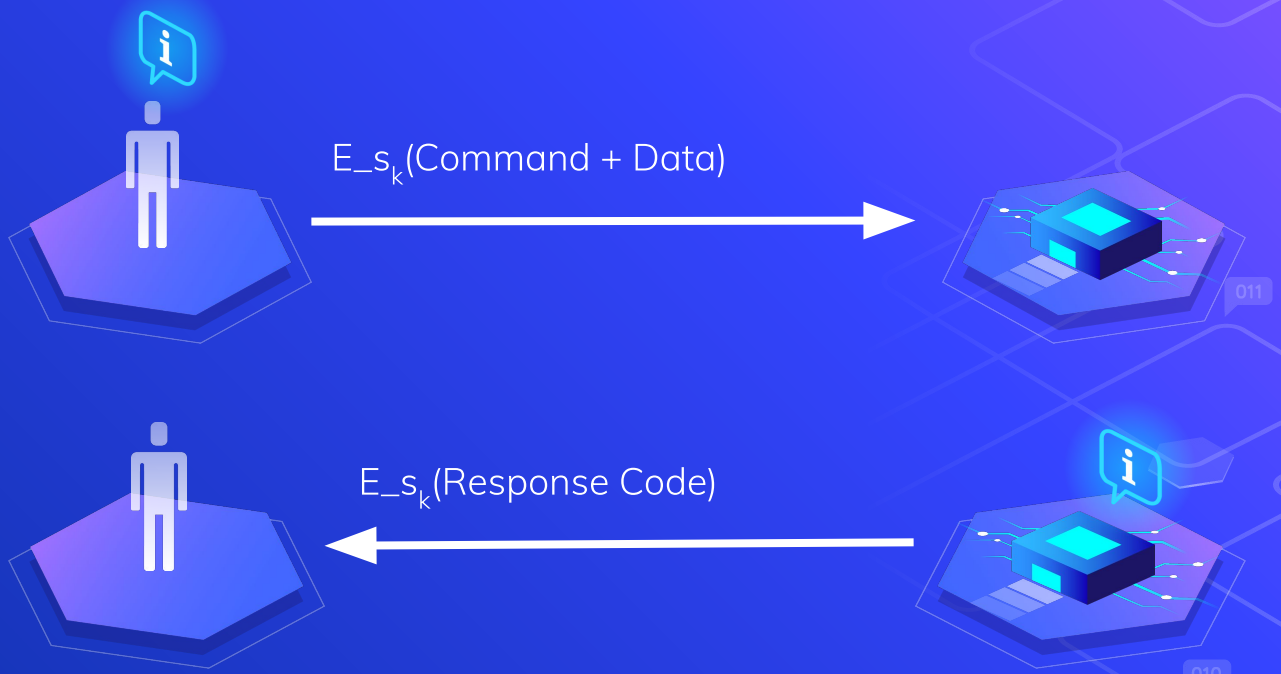
Client



Server

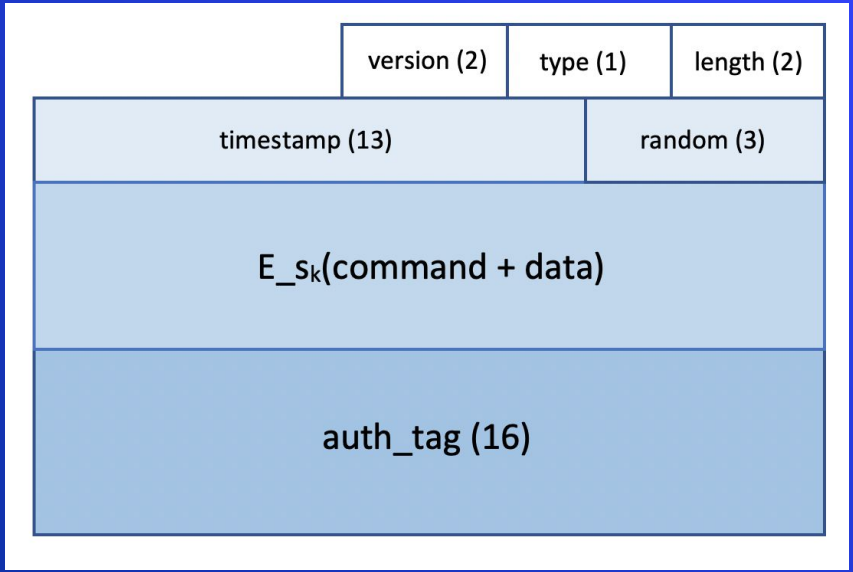


Command Protocol

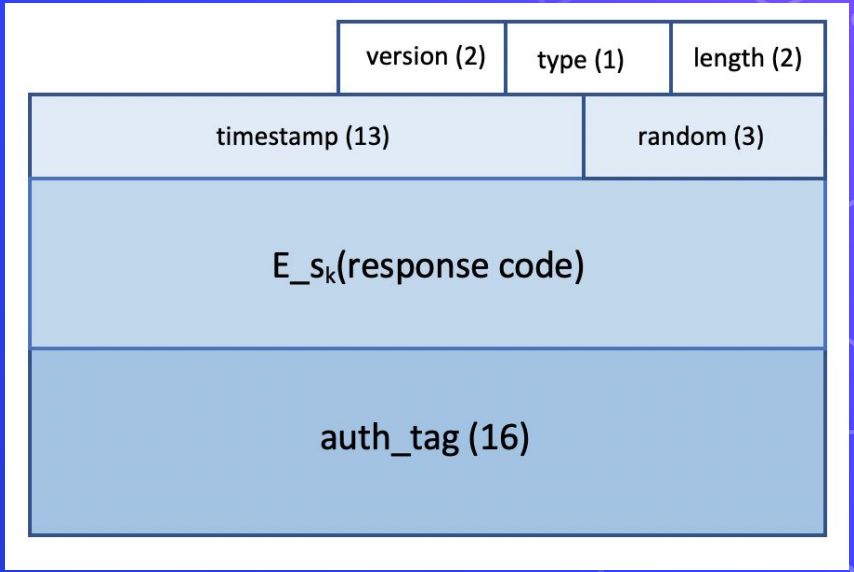


Command Protocol Format

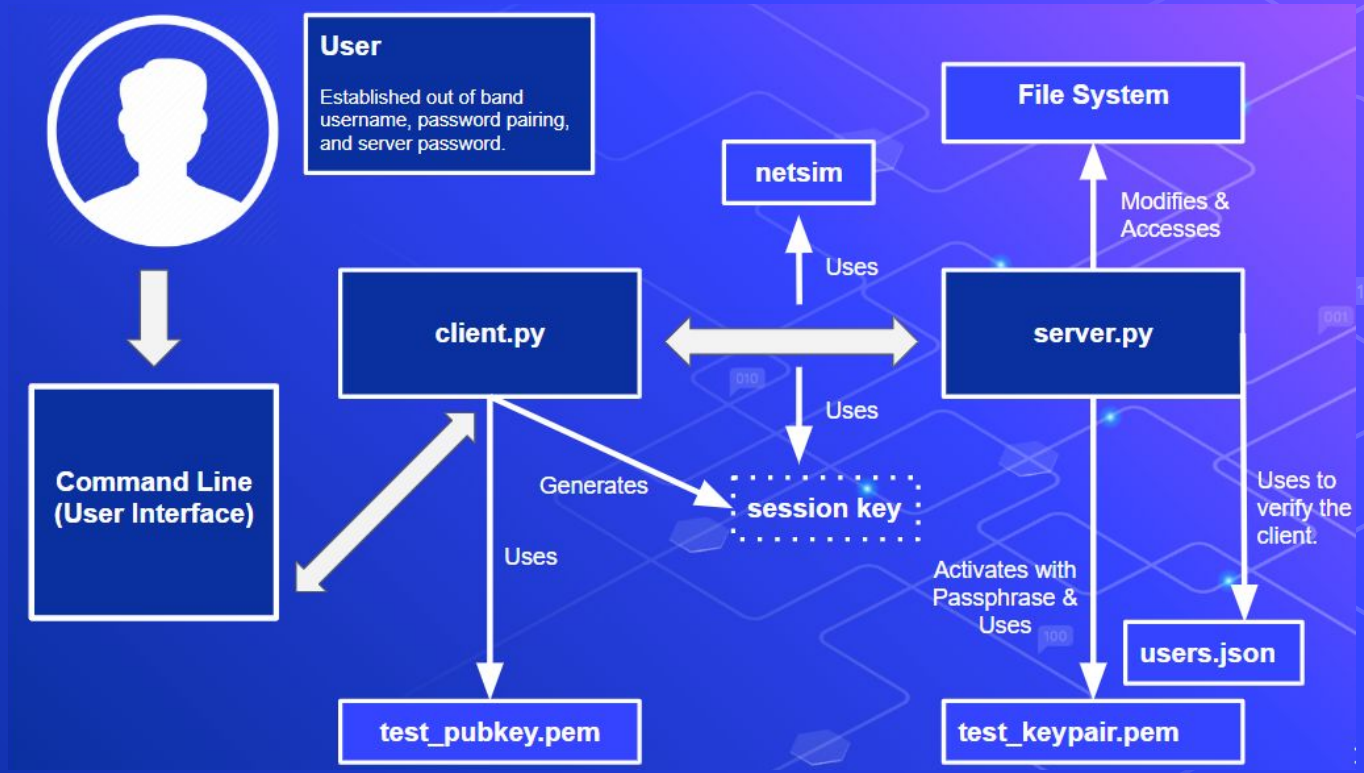
Client



Server



System Design



3/ Attack Considerations

- Attack Description
- Attack Resistance

Eavesdropping

Attack

We assume the attacker can eavesdrop on all server and client communications

Resistance

Hybrid encryption relying on fresh session key that can only be decrypted with private key, provides confidentiality.

Brute Force Attacks

Attack

We assume that attackers will try different combinations of usernames and passwords.

Resistance

Given our requirement for strong passwords and the username-password space, we have brute force attack resistance.

Replay Attacks

Attack

Attackers can replay valid messages between server and client to impersonate parties.

Resistance

The usage of timestamp and checking of timestamp freshness provide resistance against replay attacks.

Impersonation Attacks

Attack

Valid users can attempt to login into and access the contents of other users.

Resistance

We identify the netpath of the user, and ensure they do not attempt modification or commands on file systems of other users.

4/ Protocol Demo

- Functionality
- Attack Resistance

Functionality Demo

We will demonstrate the functionality of our file transfer system's login and command protocols.

Functionality Demo

Login

We simulate login as username (levente12) with an out of band user-password (ilovemath)

Folder & File Modification

We will [1] create two directories: hw and projects (MKD), [2] upload files: hw1.pdf and hw2.pdf to hw folder (UPL), [3] delete file hw1.pdf (RMF), [4], move to projects folder (CWD), [5] ask for the current folder (GWD), [6] delete the projects folder (RMD), [7] move to the hw folder (CWD), [8] list files in hw folder (LST), [9] download hw1.pdf (DNL)

Attack Demo

We will demonstrate the attack resistance of our system using attack mode.

Attack Mode Demo

Eavesdropping

All messages sent between server and client will be printed onto the command line interface.

Replay

We try to replay a valid login message, and see if the server will accept the user.

*Other attacks including field modification, session key deletion can be tested in attack mode.

Areas of improvement...

- disallow more than three login attempts for a given username
- create a registration protocol that requires strong passwords
- Implement timeout function, if client is inactive for a prolonged period, the server perceives it as an automatic logout

Pitfalls & Lessons Learned

- Security is not composable, making design challenging
- Hybrid encryption fits well into security
- Implementation was more time-consuming than expected
- Different perspectives uncover different attacks
- Much of the security we take for granted is the product of clever design and thoughtful implementation

Thanks.