[CS309] Introduction to Cryptography and Network Security

Course Instructor: Dr. Dibyendu Roy
Scribed by: Tanuj Saini (202251141)

Autumn 2024-2025
Lecture (Week 11)

1 Signal Protocol

The Signal Protocol is a modern cryptographic protocol designed for secure end-to-end communication, leveraging advanced cryptographic techniques for optimal security.

2 Key Features

The following table highlights the primary features of the Signal Protocol:

Feature	**Description**
End-to-End Encryption	Ensures only the communicating parties can read the messages, preventing third-party access.
Forward Secrecy	Guarantees that even if a session key is compromised, previous communications remain secure.
Post-Compromise Security	Ensures that communications are secure even after a compromise occurs, protecting future messages.
Double Ratchet Algorithm	Dynamically generates new session keys for each message, enhancing security and forward secrecy.

Table 1: Key Features of the Signal Protocol

3 Key Exchange Mechanics

The Signal Protocol combines **Elliptic Curve Diffie-Hellman (ECDH)** and the **Double Ratchet Algorithm** to establish secure keys.

3.1 Elliptic Curve Diffie-Hellman (ECDH) Key Exchange

The mechanism for ECDH-based key exchange is summarized below:

3.2 Double Ratchet Algorithm

The Double Ratchet Algorithm builds on the shared secret from ECDH to derive new session keys for every message. It achieves:

- **Key Derivation:** Uses a combination of hash functions and secrets to generate unique keys.

Step	Description
Key Generation	Each party generates a private key a and computes their public key $g^a \mod p$.
Public Key Exchange	Parties exchange public keys ($g^a \mod p$ and $g^b \mod p$).
Shared Secret Computation	Both parties calculate the shared secret as $K = (g^b)^a \mod p = (g^a)^b \mod p$.

Table 2: Steps in ECDH Key Exchange

- **State Updates:** Updates sender and receiver states with each message exchange.
- **Resilience:** Provides forward and backward secrecy even under message loss.

3.3 Double Ratchet Algorithm

Each party updates their keys independently with the following:

$$K'_{send} = HMAC(K_{send}, \text{nonce})$$
$$K'_{recv} = HMAC(K_{recv}, \text{nonce})$$

Where $HMAC$ is the keyed-hash message authentication code, and the keys K_{send} and K_{recv} are updated with every message.

4 Alice and Bob Example

The communication process between Alice and Bob using the Signal Protocol is summarized below:

Step	Description
Initialization	Both Alice and Bob exchange their public keys: $A = g^a \mod p, \quad B = g^b \mod p$
Shared Secret	They compute a shared secret: $S = B^a \mod p = A^b \mod p$
Message Encryption	Alice encrypts a message M using the shared secret: $C = E(S, M)$ (using AES or another symmetric encryption scheme)
Message Decryption	Bob decrypts the ciphertext to retrieve the original message: $M = D(S, C)$

Table 3: Steps in Secure Communication between Alice and Bob

5 Zero-Knowledge Proofs

The Signal Protocol uses zero-knowledge proofs (ZKPs) to verify identities without exposing private keys. The ZKP process is detailed below:

Step	Description
Commitment	Prover sends a commitment C to the verifier.
Challenge	Verifier sends a random challenge r to the prover.
Response	Prover computes and sends a response R such that: $C = H(R, r)$, where H is a cryptographic hash function.

Table 4: Steps in Zero-Knowledge Proofs

6 Applications

The Signal Protocol is widely adopted in modern secure messaging applications. Key examples include:

Application	Description
Signal Messenger	Uses the protocol as its foundation for secure communication.
WhatsApp	Implements Signal Protocol for end-to-end encryption.
Facebook Messenger	Provides secret conversations using the protocol.

Table 5: Applications of the Signal Protocol

7 Rivest Cipher (RC4)

The Rivest Cipher 4 (RC4) is a symmetric stream cipher widely recognized for its simplicity and speed. Key attributes of RC4 are summarized below:

Feature	Description
Simplicity	Easy to implement in both software and hardware.
Speed	Efficient at generating a key stream.
Stream Cipher	Operates byte-by-byte, suitable for real-time encryption.
Symmetric Key	Uses a single key for both encryption and decryption.

Table 6: Key Features of Rivest Cipher (RC4)

8 Conclusion

The Signal Protocol sets the gold standard for secure communication with innovative mechanisms like forward secrecy and post-compromise security. While RC4 has been widely used historically, its vulnerabilities have rendered it unsuitable for modern cryptographic systems.

9 RC4 Algorithm

The RC4 algorithm consists of two main components: the **Key Scheduling Algorithm (KSA)** and the **Pseudo-Random Generation Algorithm (PRGA)**. A summary of these components is provided below:

Component	Description
Key Scheduling Algorithm (KSA)	Initializes and permutes the state array S based on the input key K.
Pseudo-Random Generation Algorithm (PRGA)	Generates the key stream used for encrypting plaintext by continuously updating the state array S.

Table 7: Components of the RC4 Algorithm

9.1 Key Scheduling Algorithm (KSA)

The KSA initializes the state array S and permutes it using the key K. The steps are as follows:

1. Initialize $S[i] = i$, for $i = 0, 1, \ldots, 255$.
2. Initialize $j = 0$.
3. For $i = 0$ to 255, perform:

$$j = (j + S[i] + K[i \mod \text{key length}]) \mod 256$$

$$\text{Swap } S[i] \text{ and } S[j].$$

9.2 Pseudo-Random Generation Algorithm (PRGA)

The PRGA generates the key stream used for encrypting plaintext as follows:

1. Initialize indices $i = 0$ and $j = 0$.
2. For each plaintext byte:
   (a) Increment i: $i = (i + 1) \mod 256$.
   (b) Update j: $j = (j + S[i]) \mod 256$.
   (c) Swap $S[i]$ and $S[j]$.
   (d) Generate key stream byte: $K = S[(S[i] + S[j]) \mod 256]$.

10 Alice and Bob Example

Alice and Bob can communicate securely using RC4. The process is summarized below:

11 Applications of RC4

Although RC4 is now deprecated, it was historically used in the following applications:

Step	Description
Key Agreement	Alice and Bob agree on a shared secret key K.
Message Encryption	Alice encrypts her plaintext P using the key stream: $$C = P \oplus K_{\text{stream}}$$
Message Decryption	Bob decrypts the ciphertext C using the key stream: $$P = C \oplus K_{\text{stream}}$$

Table 8: Alice and Bob Communication Using RC4

Application	Description
SSL/TLS	Used for securing web communications in early versions of SSL/TLS.
WEP and WPA	Applied in wireless security protocols for data encryption.
File Encryption Tools	Integrated into tools for encrypting sensitive files.

Table 9: Historical Applications of RC4

12 Vulnerabilities and Deprecation

RC4 is no longer recommended due to several vulnerabilities:

Vulnerability	Description
Key Stream Biases	Early bytes of the key stream exhibit non-random patterns, making them susceptible to attacks.
Weak Key Scheduling	Certain keys result in predictable outputs, compromising security.
Susceptibility to Attacks	Vulnerable to known plaintext and statistical attacks.

Table 10: Vulnerabilities of RC4

13 Conclusion

RC4 was a significant milestone in cryptography, widely adopted for its simplicity and speed. However, due to its vulnerabilities, it has been replaced by more secure algorithms like AES. Despite its deprecation, RC4 remains an important example for understanding stream ciphers.