

Authenticated Encryption in SSH: Provably Fixing the SSH Binary Packet Protocol

Written by: Mihir Bellare, Tadayoshi Kohno,
Chanathip Namprempre
September 2002

Presented by: Khandys A. Polite
October 8, 2002

Introduction

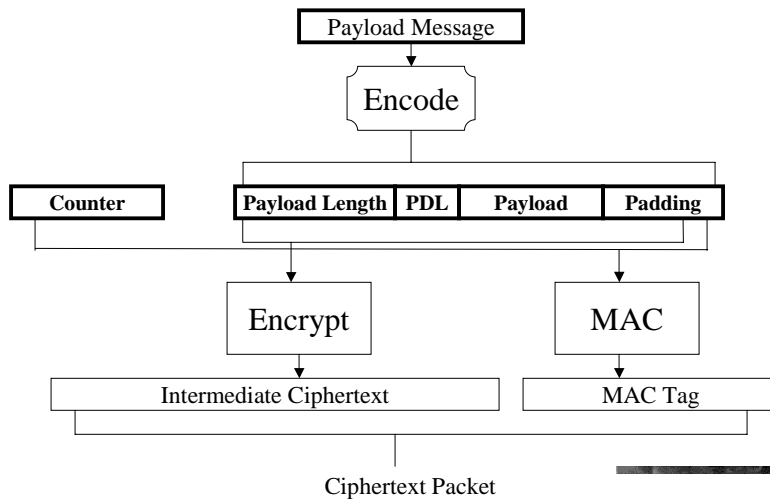
- IETF – Internet Engineering Task Force
 - www.ietf.org
 - Secure Shell (SSH) Protocol
 - Binary Packet Protocol
 - Current SSH is insecure
 - Propose several fixes
 - Provable security
-

SSH Binary Packet Protocol

- Encrypts and authenticates messages between two parties involved in an SSH connection
- Client and server agree on:
 - Set of shared symmetric keys
 - Encryption scheme (CBC)
 - Message authentication scheme (HMAC)
- SSH authenticated encryption scheme

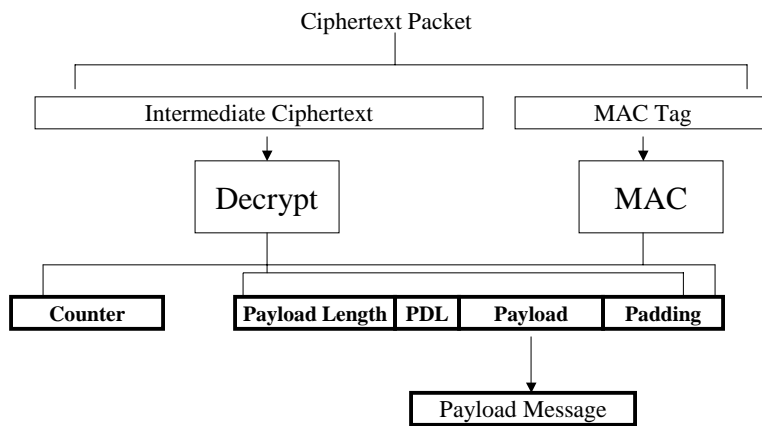
3

SSH Authenticated Encryption Scheme



4

SSH Authenticated Decryption Scheme



5

Attacks

- CBC mode encryption with chained IVs is insecure (SSH-IPC)
- CBC mode with random IVs is “provably secure” against chosen-plaintext attacks (SSH-NPC)
 - Preserves privacy as long as a user does not use it to encrypt more than 2^{32} messages with any given key
- Natural fix to use randomized CBC mode instead of chained CBC mode
- Not secure enough

6

Attacks

- Reaction Attack (SSH-NPC)
 - Attacker intercepts ciphertexts sent by a party in the SSH connection
 - Attacker makes a guess about relationship between plaintexts corresponding to intercepted ciphertexts
 - Attacker uses this guess to create a new ciphertext and sends it to other party in SSH connection
 - If second party does not accept the new ciphertext, the connection will be terminated and the attacker will know that the guess was wrong
-

7

Attacks

- Information Leakage, Replay, and Out-Of-Order Delivery Attacks
 - If an SSH-NPC or SSH-IPC session is not rekeyed frequently enough, then the session will be vulnerable to these attacks because the counter will begin to repeat causing the following:
 - *Info Leakage*: information about the plaintext will be leaked through the MAC which is nothing more than the encoded payload message prepended with the counter
 - *Replay*: once the receiver has decrypted 2^{32} messages, an attacker will be able to convince the receiver to re-accept a previously received message
 - *Out-Of-Order Delivery*: once the sender has encrypted more than 2^{32} messages, an attacker will be able to modify the order in which the messages are decrypted
-

8

Provably Secure Fixes

- Assumption:
 - these fixes are not used to encrypt more than 2^{32} packets between rekeying
 - These fixes will resist chosen-plaintext, chosen-ciphertext, forgery, replay, and out-of-order attacks
 - Randomized CBC mode encryption with random padding
 - CBC mode encryption with CTR generated IVs
 - CTR mode with stateful decryption
-

9

Provably Secure Fixes

- Randomized CBC mode with random padding (SSH-\$NPC)
 - Recall attack on SSH-NPC
 - Involved a newly created ciphertext that decrypts to an encoded packet previously encrypted by a user (if the attacker's guess was right)
 - Require the random padding be chosen in a different way for each encryption
 - Require the random padding occupy at least one full block of the encoded packet
 - Why this works:
 - Attacker will not know what the random padding is and will not be able to forge a ciphertext that will decrypt to the previously encoded message
-

10

Questions? Comments? Suggestions?
