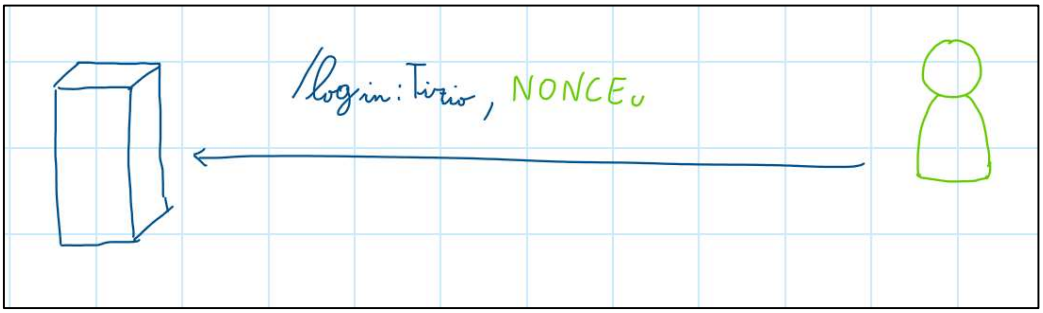
Idea realizzazione progetto Cyber security

Gruppo Topolino Hackerino: Andrea Tubak, Stefano Petrocchi

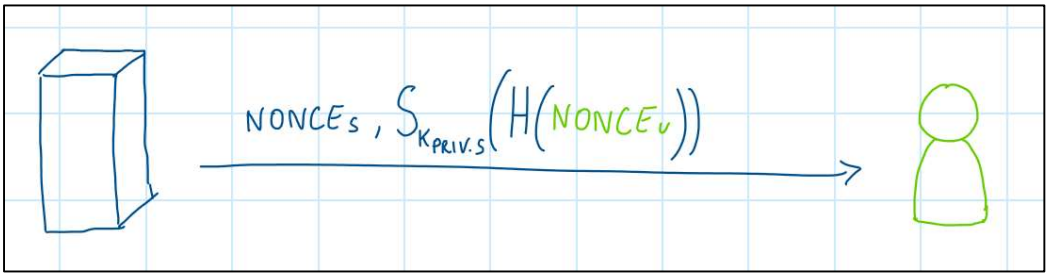
Autenticazione utente + generazione chiave di sessione

Il server possiede le chiavi pubbliche di tutti gli utenti iscritti, mentre gli utenti possiedono il certificato della chiave pubblica del server.

L'utente manda un messaggio in chiaro contenente il tipo di operazione, il nome dell'utente che si vuole loggare assieme a un numero casuale chiamato NONCEu --> /login:Nome_utente, NONCEu

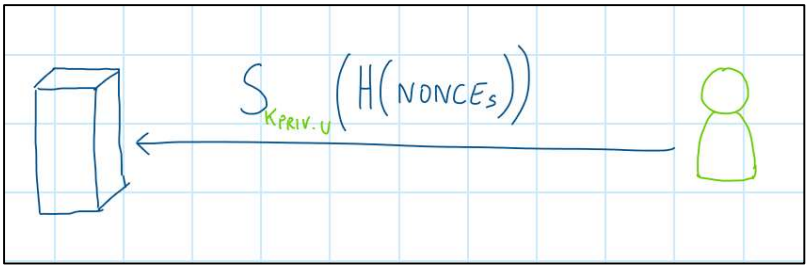


Il server controlla se l'utente può accedere e, in caso affermativo, risponde con un numero casuale chiamato NONCEs seguito dalla firma digitale di NONCEu:



Il client verifica la correttezza della firma del server con la chiave pubblica ottenuta dal certificato.

Il client termina l'autenticazione rispondendo con una firma digitale contenente NONCEs:

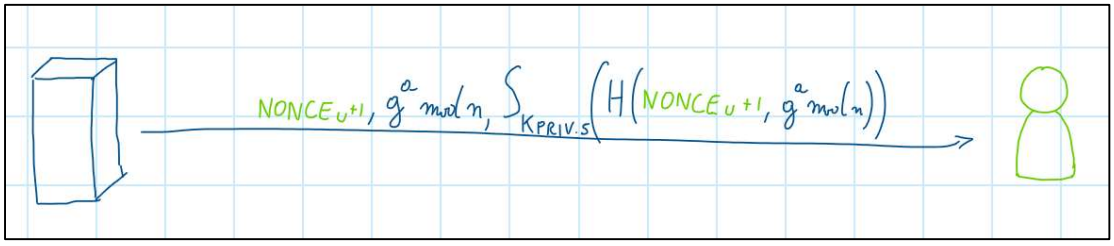


A questo punto il server può verificare l'identità dell'utente ed entrambi sapranno che i messaggi scambiati sono "freschi".

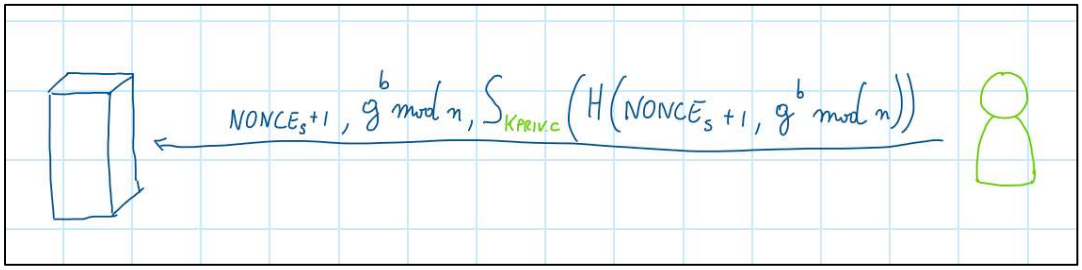
Arrivati a questo punto dobbiamo generare e scambiare una chiave di sessione per garantire la perfect forward secrecy.

La nostra idea è usare DH autenticato assieme ad un nonce incrementato ad ogni messaggio;

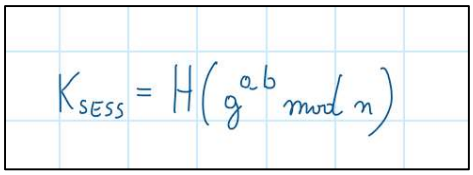
Il server genera i parametri di DH assieme alla sua chiave pubblica ed invia il seguente messaggio all'utente:



L'utente controlla la validità della firma, recupera la chiave pubblica del server e genera una sua chiave pubblica. Successivamente risponde con il seguente messaggio:

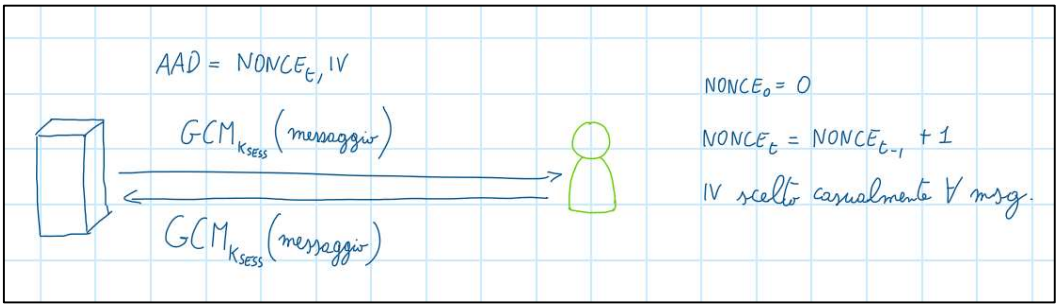


Così facendo riescono a stabilire la nuova chiave di sessione che sarà:



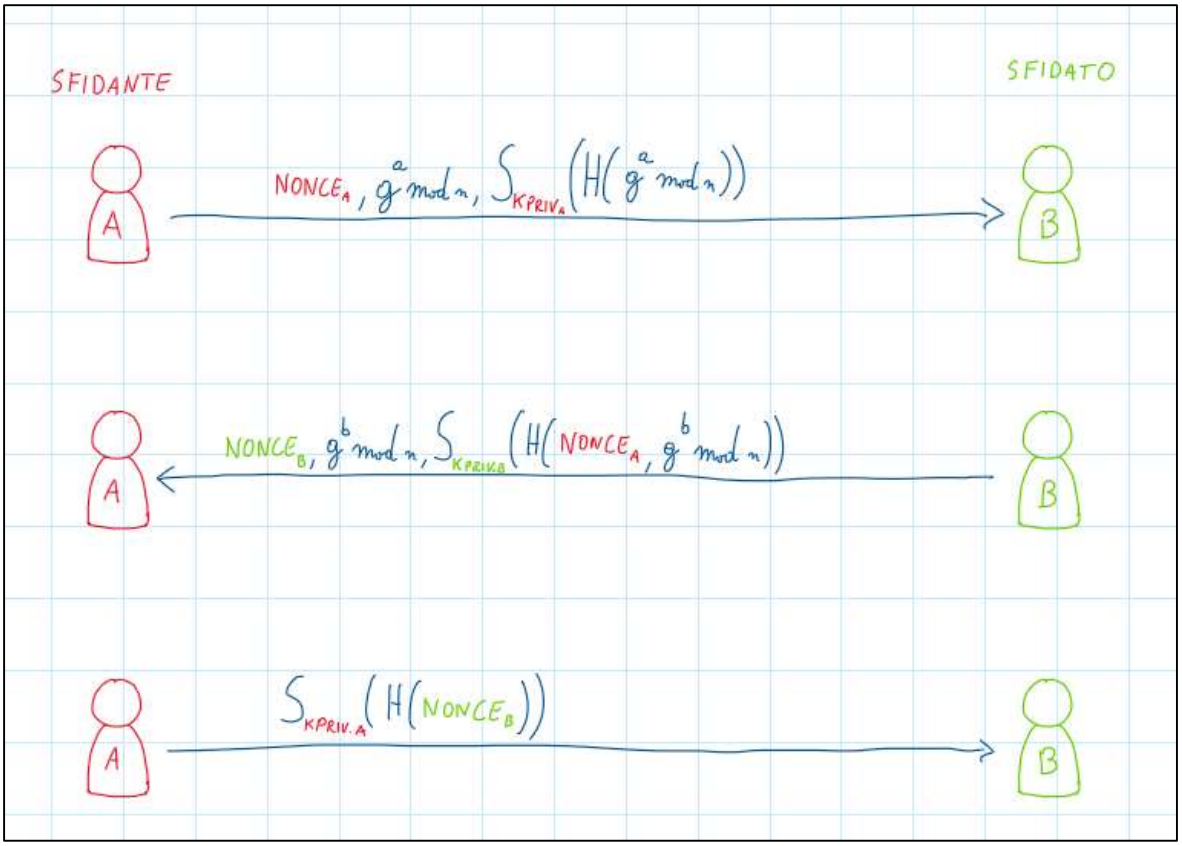
Scambio messaggi durante la sessione

I messaggi in una sessione vengono criptati e autenticati con AES-GCM. Per ogni messaggio viene inserito nell' AAD un IV casuale e un NONCE che viene incrementato di volta in volta per impedire che lo stesso messaggio venga inviato più volte. Inizialmente il NONCE parte da 0.



Autenticazione utente P2P + generazione chiave di sessione per partita

Supponiamo che il server faccia da garante per la validità delle chiavi pubbliche dei giocatori in una partita, quindi, seguendo la falsa riga di ciò che abbiamo fatto con client-server, abbiamo previsto lo scambio dei seguenti messaggi:



Alice ha sfidato Bob e Bob ha accettato. Il server ha fornito ad entrambi l'IP e la chiave pubblica dell'avversario. Alice genera un NONCEa e la nuova chiave pubblica; successivamente manda il primo messaggio.

Bob verifica la validità del messaggio, genera un NONCEb e la sua chiave pubblica; successivamente manda il secondo messaggio mostrato in figura.

Alice chiude il protocollo firmando il NONCE di Bob e appurando la freschezza e autenticità della conversazione.

La chiave di sessione tra i due client sarebbe:

$$K_{SESS.\,A-B} = H(g^{ab} \bmod n)$$

I successivi messaggi scambiati tra i client seguono uno schema identico a quello del caso client-server: