PB173 Projekt - CryMe

Aplikace **CryMe** bude sloužit k výměně zašifrovaných zpráv mezi uživateli, skrze centrální server. Hlavní rolí serveru bude správa registrovaných uživatelů, distribuce jejich veřejných klíčů a přeposílání zpráv. Zprávy půjde zasílat i když protistrana nebude připojená -- server si bude držet frontu zpráv k zaslání při dalším připojení.

Komunikace mezi klientem a serverem bude vždy probíhat šifrovaně pomocí veřejného klíče, veřejný klíč serveru bude obsažen v samotné aplikaci, aby se zabránilo útoku typu MITM. Server bude zpátky komunikovat výhradně pomocí veřejného klíče uživatele, který mu bude poskytnut informacemi v první zprávě -- buď LOGIN (tedy dohledá si patřičný klíč v databázi) nebo REGISTER (rovnou přijme nový klíč). Pro zefektivnění komunikace může být implementována funkcionalita pro dohodnutí symetrického klíče pro danou session.

Komunikace mezi klienty bude také vždy šifrovaná a to bez aktivní účasti serveru. Ten bude sloužit pouze jako prostředník pro přeposílání zašifrovaných zpráv. Klienti budou mezi sebou komunikovat pomocí veřejných klíčů. Veřejný klíč protistrany si mohou vyptat od serveru. Pro zefektnění komunikace může být implementována funkcionalita pro dohodnutí symetrického klíče mezi klienty.

Síťová komunikace bude probíhat skrze TCP spojení iniciovaná klienty. Využijeme k tomu knihovnu Asio. Samotná komunikace bude kódována vlastním jednoduchým komunikačním protokolem. K samotnému šifrování budeme využívat knihovnu mbedTLS.

Iniciální komunikace mezi serverem a klientem bude probíhat na bázi RSA2048 (v budoucích verzích možnost volby i RSA s delším klíčem či jiného asymetrického protokolu). Pomocí čtyčcestné challenge response pro ověření autenticity serveru i klienta dojde k ustanovení sdíleného symetrického klíče (na základně náhodných hodnot v challenge-response protokolu). Bitové reprezentace náhodných hodnot, které si vyměnili, budou zřetězeny a zahashovány SHA2-256 pro vytvoření symetrického klíče pro AES-256 se zachováním entropie z obou vstupů. Další komunikace mezi serverem a klientem bude šifrována AES-256.

Pro komunikaci mezi samotnými klienty bude využit asymetrický protokol dle jejich klíče (pro začátek RSA2048). V případě delší komunikace může dojít k vytvoření symetrického klíče podobně jako u komunikace server-klient.

Server si pamatuje veřejné klíče jednotlivých registrovaných uživatelů - drží databázi dvojic (pseudonym, klíč). V případě trvajícího přihlášení i TCP kanál na komunikaci s nimi (přes který se přihlašovali) a domluvený symetrický klíč. V případě LOGOUT() ukončí spojení a smaže symetrický klíč.

Komunikační protokol

Definujeme jednoduchý komunikační protokol mezi klienty a serverem.
První bajt zprávy popisuje její typ a následující bajty budou interpretovány dle typu zprávy.
Posledních 32 bajtů zprávy bude hash pro ověření integrity.

Server s nově příchozím spojení počítá, že první dvě zprávy (LOGIN nebo REGISTER a následná RESPONSE na challenge) budou zašifrovány jeho veřejným klíčem. Během této komunikace dojde k výměně informací pro dohodnutí nového symetrického klíče pro šifrování dalších zpráv daného spojení.

Druhy zpráv:

- REGISTER(pseudonym, pub_key) - iniciuje challenge response protokol pro ověření znalosti privátního klíče a následného zaregistrování klienta (uložení si pseudonymu a pub_key)
- LOGIN(pseudonym) - iniciuje challenge response protokol na oveření znalosti privátního klíče
- CHALLENGE(challenge) - zašle výzvu
- RESPONSE(response) - zašle odpověď
- SEND_TO(pseudonym, message) - zašle zprávu uživateli (od odesilatele serveru)
- RECEIVE_FROM(pseudonym, message) - zašle zprávu uživateli (od serveru příjemci)
- REQUEST_KEY(pseudonym) - odesílatel žádá o veřejný klíč uživatele
- RETURN_KEY(pseudonym, pub_key) - server zasílá veřejný klíč uživatele
- LOGOUT() - uživatel chce ukončit spojení
- ASK(pseudonym, pub_key2) - žádost od serveru příjemci (s pub_key1), že s ním chce navázat odesílatel (s puk_key2) spojení (odesílatel požádal o jeho pub_key1)

Z pohledu uživatele

- může si vygenerovat dvojici klíčů
- zaregistrovat se - poslat žádost serveru se svým pseudonymem a věřejným klíčem
- požádat server o spojení s jiným uživatelem
- posílat a přijímat zpávy, komunikovat s jiným uživatelem
- ukončit spojení se serverem
- pamatuje si uživatele, se kterými komunikoval (pseudonym, veřejný klíč)