UNIVERSITA' DEGLI STUDI DI CAGLIARI
FACOLTÀ DI SCIENZE
Corso di Laurea in Informatica

Analisi della libreria crittografica PyCryptoDome

ANNO ACCADEMICO 2022-2023

Docente di riferimento

Prof. Massimo Bartoletti

Candidato

Michele Melis (matr.65798)

Indice

- Introduzione
- Algoritmi di hashing e schemi di autenticazione
- Schemi di cifratura a chiave privata
- Schemi di cifratura a chiave pubblica
- Schemi di firma digitale
- Protocolli di condivisione e funzioni di derivazione chiavi
- Conclusioni

Introduzione

L'elaborato pone il focus sull'analisi e l'utilizzo delle primitive crittografiche implementate nella libreria **PyCryptoDome** per la realizzazione di script esemplificativi a seconda del caso d'uso preso in esame.

La libreria è suddivisa in otto package e, per ogni package, vengono implementate le seguenti primitive/funzioni:

- **Hash**: algoritmi di hashing e schemi di autenticazione;
- **Random**: funzioni per la generazione di numeri pseudo-casuali;
- **IO**: formati di memorizzazione chiavi;
- **Cipher**: schemi di cifratura a chiave privata;
- **PublicKey**: schemi di cifratura a chiave pubblica;
- **DigitalSignature**: schemi di firma digitale;
- **Protocol**: funzioni di derivazione chiavi e protocolli di condivisione;
- **Util**: funzioni accessorie per lo svolgimento di operazioni secondarie.

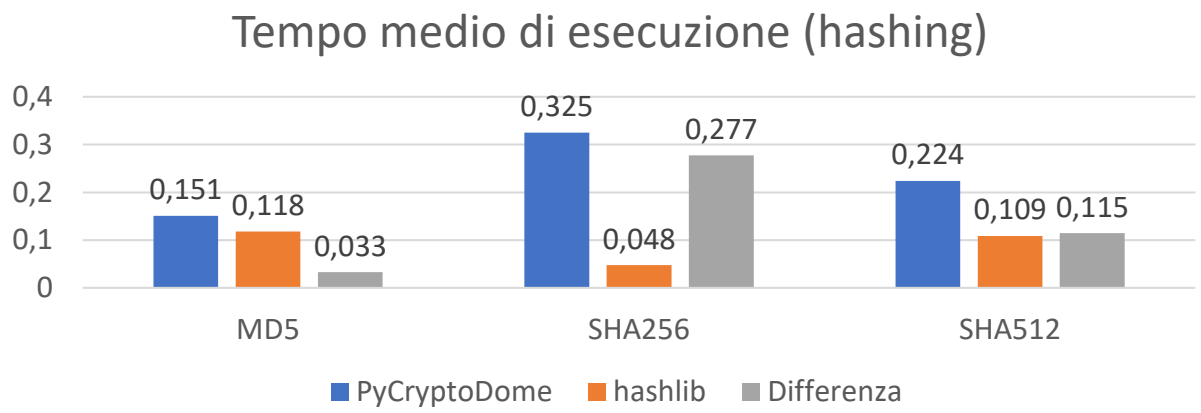
Per ogni primitiva utilizzata negli script ed implementata in almeno un altro modulo o in un'altra libreria, viene eseguito un confronto sui tempi medi di esecuzione tramite test t con significatività 1%. Le metriche utilizzate sono riportate nell'elaborato.

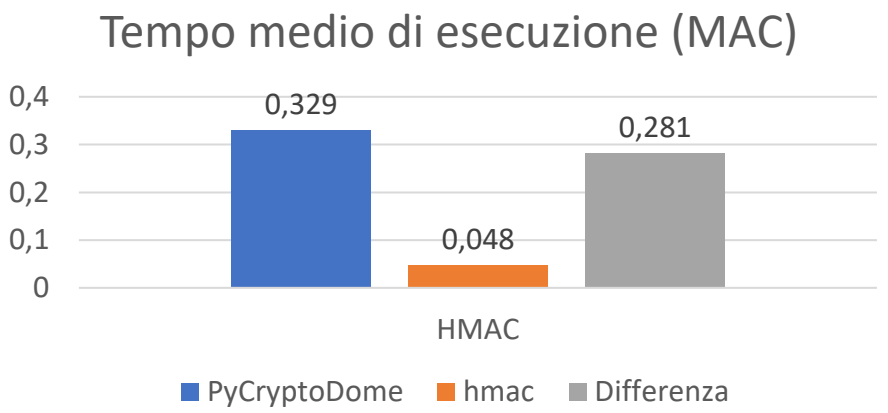
Algoritmi di hashing e schemi di autenticazione

Primitive implementate nel package: **SHA-2**, **SHA-3**, **BLAKE2**, **HMAC**, **CMAC** e **Poly1305**.

Casi d'uso presi in esame:

- **garantire la sicurezza di una password** (con SHA512);
- **controllo integrità dati** (con MD5 e SHA256);
- **sistema di autenticazione di un messaggio** (con HMAC).



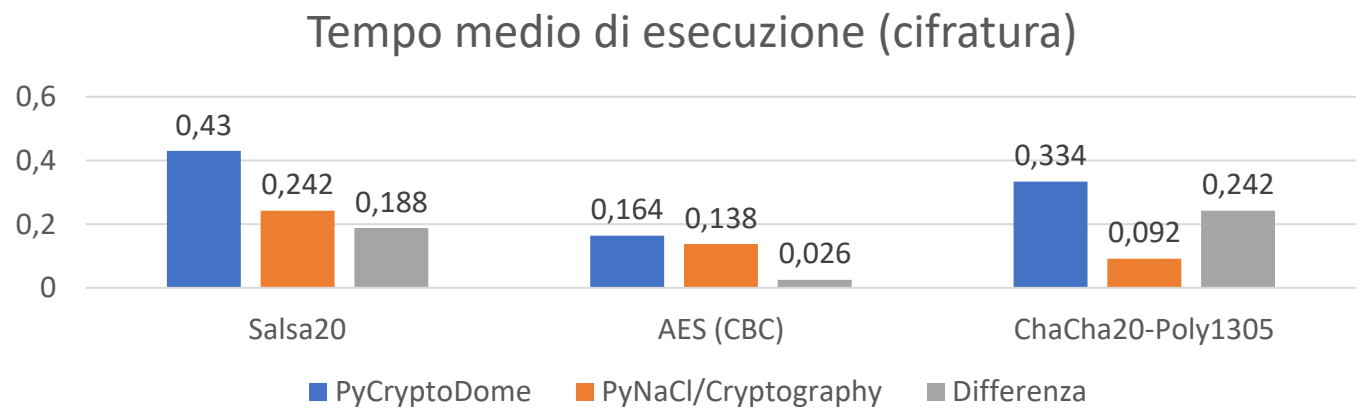


Schemi di cifratura a chiave privata

Primitive implementate nel package: **Salsa20**, **ChaCha20**, **XChaCha20**, **AES**, **PKCS#1-OAEP**.

Casi d'uso presi in esame:

- **cifratura di un messaggio testuale tramite cifrario a flusso** (con Salsa20)
- **cifratura di un file tramite cifrario a blocco** (con AES CBC mode)
- **cifratura messaggio con generazione codice di autenticazione** (con ChaCha20-Poly1305)

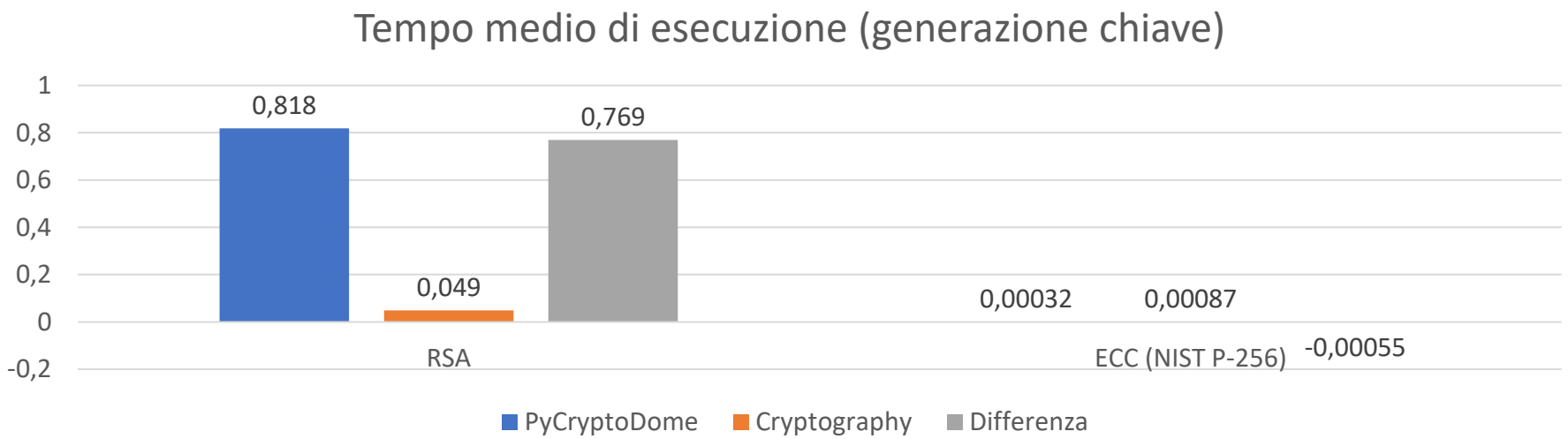


Schemi di cifratura a chiave pubblica

Primitive implementate nel package: **RSA**, **DSA**, **ECC** ed **ElGamal**.

Casi d'uso presi in esame:

- **condivisione chiave di sessione** (con RSA)

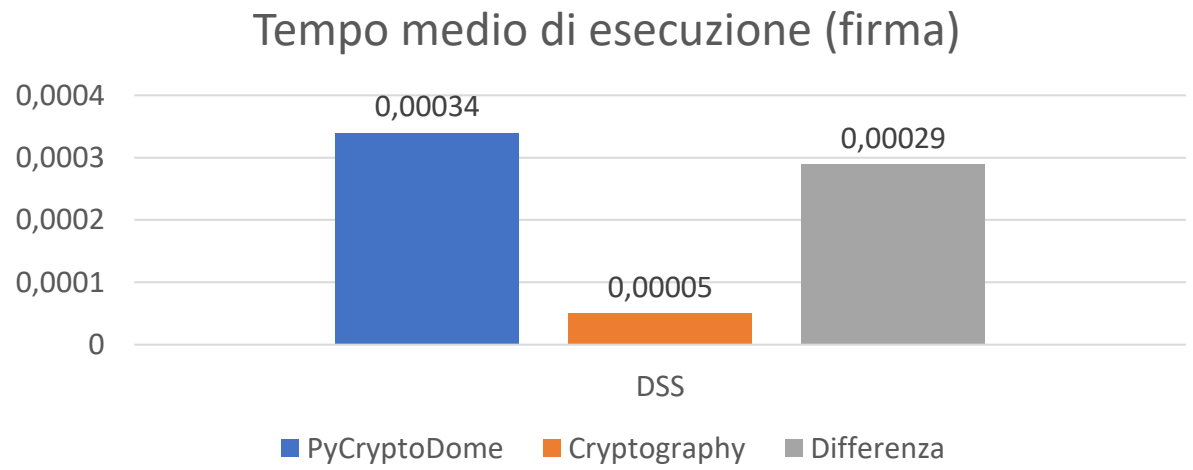


Schemi di firma digitale

Primitive implementate nel package: **PKCS#1 v1.5**, **PKCS#1 PSS**, **DSA**, **EdDSA** e **ECDSA**.

Casi d'uso presi in esame:

- **firma di un documento digitale** (con PKCS#1 PSS)
- **sistema di autorizzazione tramite certificato** (con ECDSA)

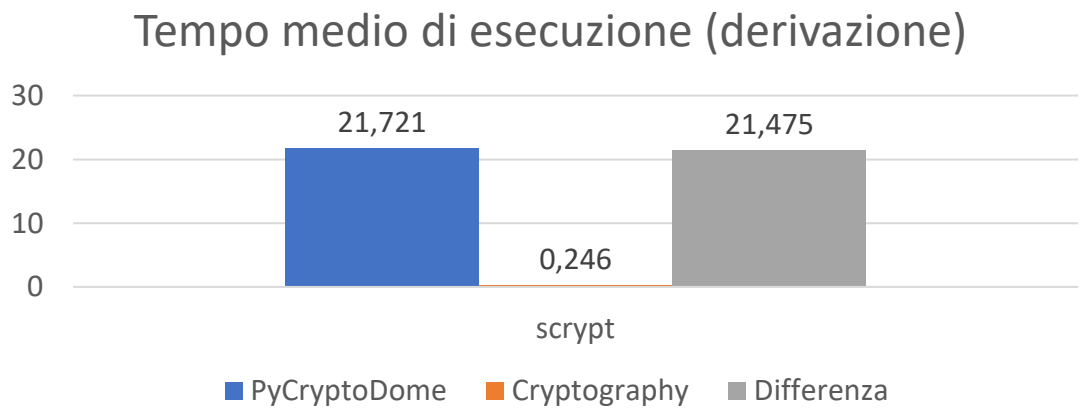


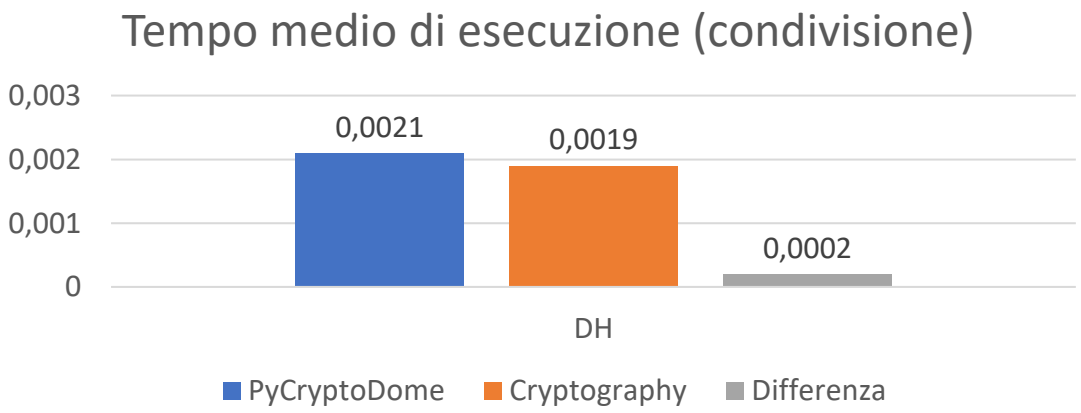
Protocolli di condivisione e funzioni di derivazione chiavi

Primitive implementate nel package: **Shamir's secret sharing**, **Diffie-Hellman** e le funzioni di derivazione chiave: **PBKDF1**, **PBKDF2**, **scrypt**, **bcrypt**, **HKDF** e **SP 800-180**.

Casi d'uso presi in esame:

- **garantire sicurezza a dati sensibili** (con scrypt)
- **accesso privato a un messaggio condiviso** (con Shamir's secret sharing)
- **condivisione chiave di sessione** (con ECDH)





Conclusioni

In conclusione, possono essere elencati i pregi e i difetti trovati durante l'analisi della libreria.

Pregi:

- Primitive: la libreria implementa un gran numero di primitive, rendendola utile per lo svolgimento diversificato di un gran numero di operazioni crittografiche (e non);
- Sintassi: la libreria offre una sintassi semplice, favorendone la sua implementazione e la successiva manutenzione del codice;
- QoL: la libreria viene manutenuta e aggiornata con regolarità, aggiungendo primitive e funzioni al passo con gli standard correnti;
- Retrocompatibilità: la libreria può essere utilizzata con standard deprecati ed è compatibile con i sistemi che utilizzando la libreria (deprecata) **PyCrypto**;

Difetti:

- Efficienza: la libreria ha dei tempi di esecuzione maggiori rispetto alla concorrenza.

Grazie per l’attenzione!