Заштита информација 2025/2026.

Поштовани студенти,

У оквиру овог пројекта реализоваћете апликацију која омогућава кодирање, декодирање, праћење и размену различитих типова датотека коришћењем криптографских алгоритама. Циљ пројекта је да покажете познавање криптографских алгоритама, рада са фајл системом, мрежном комуникацијом, обрадом података и добрим принципима развоја софтвера.

Избор алгоритама и формирање тимова

- Имплементираћете по један алгоритам из сваке групе дате у *Table 1*. Корисник апликације бира којим ће се алгоритмом кодирати фајлови. Механизам за дистрибуцију кључа је произвољан.
- Потребно је да сваки студент самостално развије свој систем.
- Сваки студент треба да пронађе још једног или двоје студената који имају исте алгоритме и да са њима формира мини-тим за демонстрацију. Тимови се формирају искључиво ради демонстрације, сваки студент самостално прави апликацију и сваки студент се независно оцењује.
- На одбрани пројекта заједно демонстрирате реалну размену фајлова путем TCP сокета, тако да један студент шаље, а други преузима и верификује фајл.

Студент имплементира алгоритам под редним бројем ***број индекса % 10 + 1***. За имплементацију погрешног алгоритма не добијају се поени.

Table 1: Алгоритми за кодирање

Редни број	Група 1	Група 2	Група 3 - *mod* за групу 2	Група 4 - критографски хеш
1	Railfence cipher	XXTEA	CBC	Tiger hash
2	RC4	XTEA	CBC	BLAKE
3	Playfair cipher	RC6	PCBC	SHA 1
4	Foursquare cipher	LEA	PCBC	SHA 2
5	Double transposition	A5/2	CFB	MD 5
6	Enigma	XXTEA	CFB	Tiger hash
7	A5/1	XTEA	OFB	BLAKE
8	Bifid	RC6	OFB	SHA 1
9	TEA	LEA	CTR	SHA 2
10	Simple substituion	A5/2	CTR	MD 5

Metadata header

Пре кодирања садржаја фајла, неопходно је да апликација генерише ***metadata header*** (у *JSON* или *XML* формату) који садржи:

- оригинални назив фајла са екстензијом,
- величину фајла,
- датум и време креирања,
- примењени алгоритам кодирања,
- примењени алгоритам за хеширање (уколико постоји).

Овај *header* се уписује у посебан сегмент фајла пре кодирања и мора да се чита на страни примаоца, током демонстрације.

File System Watcher (FSW)

Потребно је омогућити коришћење *FSW* механизма за аутоматску детекцију додавања нових фајлова у изабрани директоријум (*Target*). Када је *FSW* укључен:

- сваки нови фајл се аутоматски детектује,
- апликација приказује поруку кориснику о томе да је фајл откривен,
- врши се аутоматско кодирање фајла,
- креира се лог запис о детекцији и кодирању.

Када је *FSW* искључен, корисник ручно бира који ће фајл бити кодиран.
Сви кодирани фајлови се чувају у директоријуму X.

Размена датотека путем TCP сокета

Када је укључена опција за размену датотека потребно је повезати се са апликацијом другог студента који је имао исте алгоритме за кодирање. Страна која шаље податак треба одабрати фајл и над њим извршити кодирање. Поред кодирања садржаја фајла, потребно извршити и одговарајући криптографски хеш алгоритам којим ће прималац проверити да ли је фајл исправно пренет. Страна која прима податак треба проверити да ли је фајл исправно пренет и ако јесте, аутоматски извршити декодирање примљеног фајла. Резултат декодирања треба бити фајл који је послат.

Логовање свих активности

Апликација мора да води евиденцију и логује активности о:

- покретању и заустављању FSW-а,
- детекцији нових фајлова,
- кодирању и декодирању,
- слању и пријему датотека,
- верификацији хеша.

Подршка за различите типове фајлова, укључујући велике фајлове

Пројекат треба да подржи:

- текстуалне фајлове (txt, docx, pdf),
- бинарне фајлове,
- слике велике резолуције,
- видео фајлове већих димензија.

Решење: Своје решење предајете као ZIP архиву у оквиру креираног задатка на каналу предмета. Назив архиве мора да садржи број индекса.

Технологија: Можете користити било који програмски језик/окружење за имплементацију, осим програмског језика Python.

Алгоритми: Све алгоритме сами имплементирате.

Дизајн апликације: Када направите дизајн своје апликације, запитајте се „да ли би ово неко користио ако изгледа овако како изгледа“. Код треба да буде написан тако да се избегне дуплирање кода, хардкодирање константи и са минималном цикломатском сложеношћу.

Оцењивање: Оцењиваћемо решење које предајете у предвиђеном року, а одбрана пројекта ће се накнадно организовати.

Референце:

Група 1:

1. Railfence cipher (https://web.archive.org/web/20120105152732/http://cryptogram.org/cdb/aca.info/aca.and.you/chapter_09.pdf#RAILFE)
2. RC4
3. Playfair cipher (https://www.geeksforgeeks.org/playfair-cipher-with-examples/)
4. Foursquare cipher (http://practicalcryptography.com/ciphers/four-square-cipher/)
5. Double transposition
6. Enigma (http://practicalcryptography.com/ciphers/mechanical-era/enigma/)
7. A5/1
8. Bifid (http://practicalcryptography.com/ciphers/classical-era/bifid/)
9. TEA
10. Simple substituion

Група 2:

1. XXTEA (https://en.wikipedia.org/wiki/XXTEA)
2. XTEA (https://en.wikipedia.org/wiki/XTEA)
3. RC6 (https://en.wikipedia.org/wiki/RC6)
4. LEA (https://en.wikipedia.org/wiki/LEA_(cipher))
5. A5/2 (https://medium.com/@shubhamkatheria11/a5-2-ciphering-algorithm-implementation-d594abd06ab8)
6. XXTEA (https://en.wikipedia.org/wiki/XXTEA)
7. XTEA (https://en.wikipedia.org/wiki/XTEA)
8. RC6 (https://en.wikipedia.org/wiki/RC6)
9. LEA (https://en.wikipedia.org/wiki/LEA_(cipher))
10. A5/2 (https://medium.com/@shubhamkatheria11/a5-2-ciphering-algorithm-implementation-d594abd06ab8)

Група 3 – mod за групу 2

(https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation)