University of South-Eastern Norway

5 November, 2024

BTS4410-1 24H Sikkerhet og Kryptografi - Oblig 3

Daniel Hao Huynh, 276562

Introduksjon

Dokumentere og programmere en versjon av SUCI funksjonaliteten. Det vil si, dette er inspirert av ECIES/SUCI, men avviker en del fra SUCI slik vi kjenner det fra 5G. Vi skal f.eks. bruk «ConcatKDF» som vår «key derivation function». Videre vil vi bruke AEAD (AES-GCM) for beskyttelsen av identifikatoren.

Innhold

Implementasjon

Jeg har implementert filene basert på tildelte filer fra Geir Køien[1]:

- deconceal.py - for selve deconceal funksjonen
- extract_tests.py - for lett måte å hente tester på
- test_Home.py - for å unitteste deconceal()

Dette er min implementasjon[2] av en SUCI funksjonalitet, Deconceal funksjonalitet i tillegg til testing av Deconceal. funksjonelitetene er da implementert modulært slik at det har vært mulig å implementere alt på en oversiktlig og testbar måte.

Oppsett av prosjekt

1. sett opp virtuell miljø:

- https://docs.python.org/3/library/venv.html

2. aktiver miljø

```
.\.venv\Scripts\activate
```

3. hent nødvendige moduler:

```
pip install -r .\requirements.txt
```

Nødvendige flags

Generering av Private og Public key

```
python Home.py "keygen"
```

Conceal brukerinformasjon

```
python User.py "conceal"
```

Deconceal brukerinformasjon

```
python Home.py "deconceal"
```

Sammenligning med andre

Her er unittesten kjørt med ECDH_PUBLIC_KEY.PEM og SUCI_data.bin tilsendt fra medstudent Patryk Okupski, 236616

```
1 Test files extracted in ./tests/*
Testing folder: Patryk
    SUCI_data written to file. Len: 98
        IV:b'>\xfd\xbf\x8d\x9a\xba\xda5FG\xadUQ\x966\x83'
        Home ID:sidf@home.org
        User ID:Patryk-Okupski

.
----------------------------------------------------------------------
Ran 1 test in 0.117s

OK
```

Unit tester

for å reprodusere resultatene, tast inn fra rot:

```
python -m unittest
```

```
4 Test files extracted in ./tests/*
Testing folder: Patryk
    SUCI_data written to file. Len: 98
        IV:b'>\xfd\xbf\x8d\x9a\xba\xda5FG\xadUQ\x966\x83'
        Home ID:sidf@home.org
        User ID:Patryk-Okupski


Testing folder: TEST_SET_0
    SUCI_data written to file. Len: 110
        IV:b'\x92=9\x1ba\xfbu\x1f\x1b3e\x8e\xd1\xaat\xbe'
        Home ID:sidf@home.org
        User ID:privacy-sensitive-name ÆØÅ


Testing folder: TEST_SET_1
    SUCI_data written to file. Len: 117
        IV:b'\x97\x84:\xf3\x17\x08(A\xfe\xadM\xba\x81\xb9\\\xcc'
        Home ID:sidf@home.org
        User ID:privacy-sensitive-name ÆØÅ


Testing folder: TEST_SET_2
    SUCI_data written to file. Len: 122
        IV:b"\x9f\xa5\x87\x8f\xf4\x8f\xa3\x86\xb7'6q\x8f\x8e\xba\x1c"
        Home ID:sidf@home.org
        User ID:privacy-sensitive-name ÆØÅ


.
----------------------------------------------------------------------
Ran 1 test in 0.117s

OK
```

Bibliografi

[1] G. Køien, «BTS4410 – SUCI implementation presentation». oktober 2024.

[2] Mystodan, «Mystodan/BTS4410-Obligatorisk-oppgave-3». Åpnet: 5. november 2024. [Online]. Tilgjengelig på: https://github.com/Mystodan/BTS4410-Obligatorisk-oppgave-3