Biztonságos platforma üzenetek továbbítására egy multicast hálózatban

Secure platform for exchanging messages in a multicast environment

DOBAI Zsolt, Dr. GENGE Béla

University of Medicine, Pharmacy, Sciences and Technology of Târgu Mureș,
Romania e-mail: dobai.zsolt@stud18.umfst.ro, bela.genge@umfst.ro

Abstract

Multicast platforms are infamous for their difficulty in providing solid security. Since messages come from many different sources, it becomes a real challenge figuring out the legitimacy of these messages. Our multicast platform demonstrates how, with a couple of cryptographic tools and methods, the security of these platforms can be improved against cyber-attacks.

Kivonat

Multicast hálózatok híresek arról, hogy nehezen kezelhetőek digitális biztonság szempontjából. Információk indulnak és érkeznek sok pontból és emiatt gyakran nehéz biztosítani az üzenetek autenticitását. Kídolgozott multicast platformánk bemutat kriptografikus módszereket melyek segítségével növelhetjük rendszerünk biztonságát kibertámadások ellen.

Kulcsszavak: multicast hálózat, kriptográfia, információ kűldés, digitális biztonság

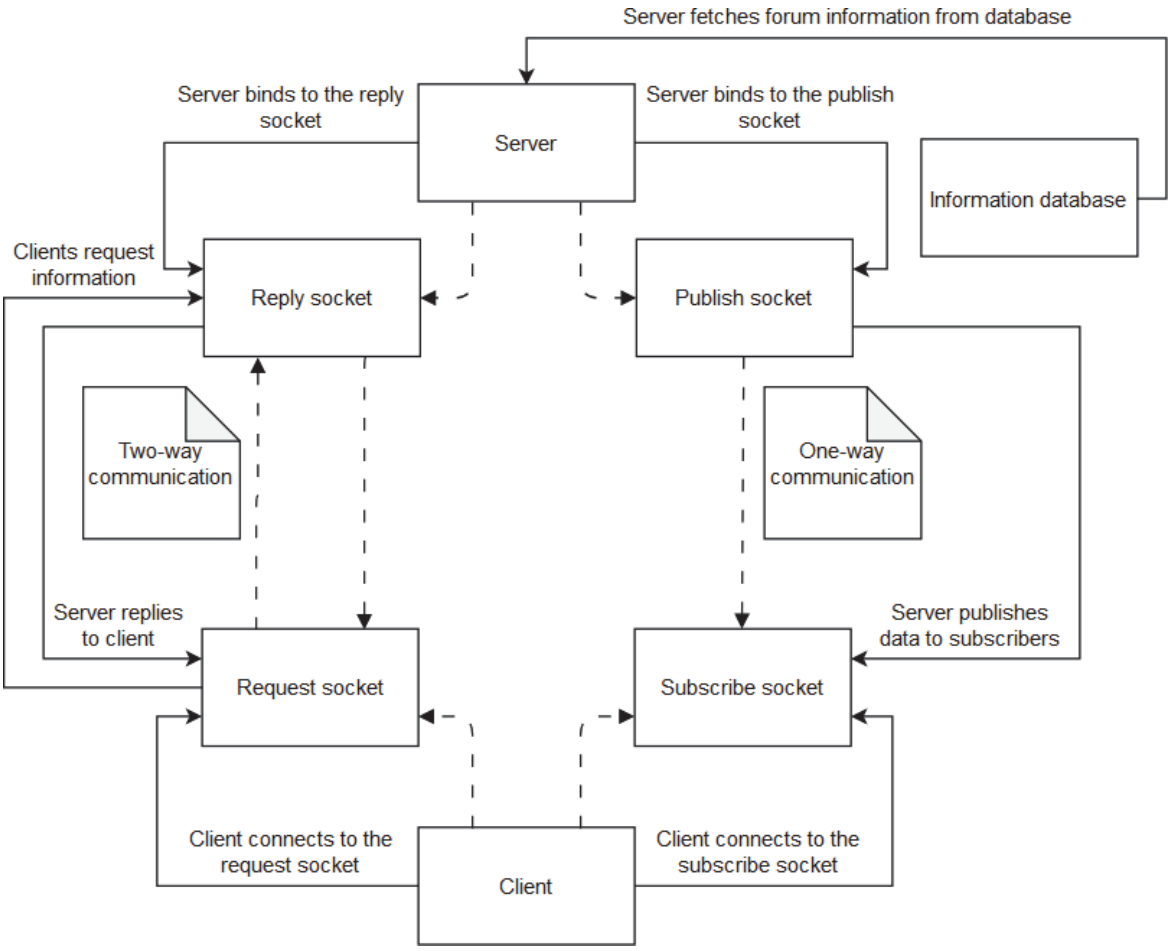
1. BEVEZETŐ

A digitális biztonság eléréséhez olykor komoly problémákkal kell megküzdenünk, hiszen a támadóknak számtalan eszköz áll a rendelkezésükre a rendszerek feltöréséhez. Támadások, mint a közbeékelődéses támadás vagy a visszajátszás támadás komoly problémákat tudnak okozni, olykor áttörni komplex rendszereken és hozzáférni kritikus információhoz következtetésképpen.

[1] Közli azt, hogy a sok kriptografikus eszköz, ami a rendelkezésünkre áll sajnos gyakran nem elegendő komoly biztonság eléréséhez. Ezért gyakran előfordul az, hogy egy rendszer képes megfékezni egy bizonyos fajta támadást, viszont kiszolgáltatottá válik más fajta támadásoktól.

A komplexitás tovább nő multicast hálózattok esetében. Kriptografikus kulcsok cseréje, üzenetek aláírása és protokollok bevetése komplikálttá válik amikor nincs egy megbízható fél.

Emiatt fontos megemlíteni azt, hogy a kidolgozott rendszerünk áttörhető, mint akármilyen rendszer, viszont a bevetett protokollok és kulcsok miatt a támadók nem férnek hozzá a teljes információhoz. Következtetésképpen, a fő célunk az volt, hogy az adatok biztosításánál mind a két fél (az adó és a vevő) vegyen részt, emiatt egyikük sem tett szert túl sok információra.



1. ábra
Rendszernek a komponensei és kapcsolata

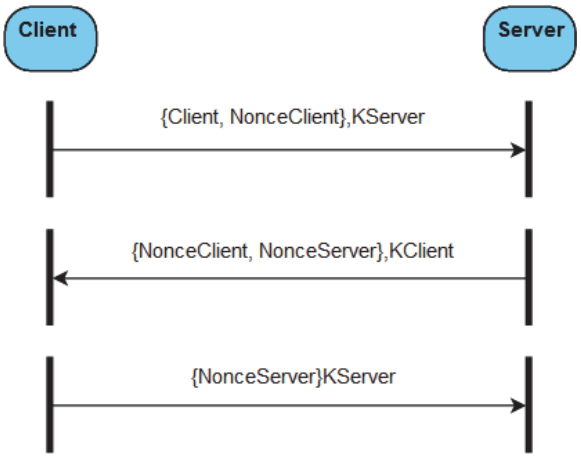
Az 1 ábra bemutatja vázolva a rendszerünk kapcsolatait. Képesek voltunk csoport kommunikációra, valamint egyszerű, kétfeles kommunikációra a szerverünkkel, az információ lekéréséhez.

[2] szerint, a csoport kommunikáció esetében 3 dologra kell figyelni: bizalmasság a résztvevők között, az üzenetek integritására, valamint a csoport bizalmasságára. Az utolsót nagyon nehéz megvalósítani, mivel az információ olykor sok ismeretlen félhez kell, hogy eljusson.

2. FELHASZNÁLT KRIPTOGRAFIKUS ESZKÖZÖK ÉS PROTOKOLOK

Az adatok titkosításához használtunk szimmetrikus és aszimmetrikus kriptálást, valamint az üzenetek aláírásához MAC-et használtunk (Message Authentication Code). Ezek az eszközök lehetővé tették, hogy biztosított adatokat, valamint autentifikált üzeneteket küldjünk a kulcsok segítségével.

A kulcsok tárolva voltak a felhasználók rendszerében a felismerés érdekében. Ez a SecureShell protokoll alapelve. [3] közli hogyan lehet felhasználni publikus kulcsokat autenticitásra.



2. ábra
Kölcsönös autentifikálás – Needham Schroeder Protokol

[4] tanulmányozva észre lehet venni a hatékonyságát ennek a protokollnak. Hatékony közbeékelődéses támadások léteznek erre a protokollra, viszont az alapelv nagyon hasznos arra, hogy biztosítsuk a résztvevők autenticitását.

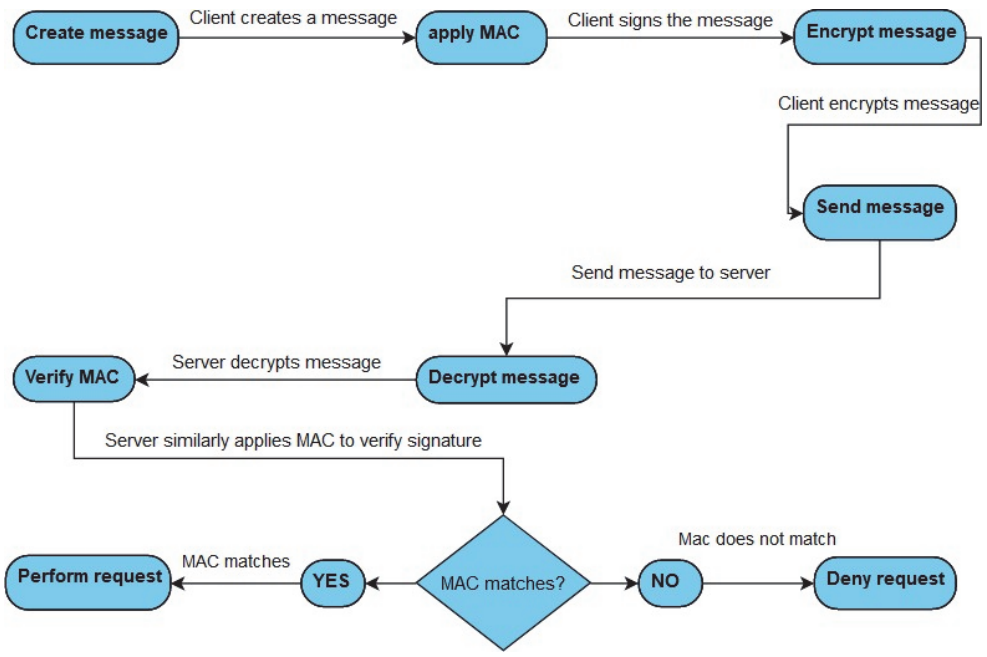
Észrevehető a 2 ábrán ahogyan, felhasználva egy közösen ismert titkot, (ebben az esetben a két résztvevőnek a publikus kulcsát) véghez lehet vinni egy üzenetcserét, aminek a végeredménye a kölcsönös autenticitás. A részvevők ezek után képesek voltak elküldeni érzékenyebb rendű információt.

A multicast kommunikáció esetében létrehoztunk csoportokat, ahol a tulajdonosok feleltek csoportjuk biztonságáért. Minden csoport rendelkezett egy szimmetrikus kulccsal, így képesek voltak titkosítani adatokat, amiket csak a tagok tudtak elolvasni.

Mivel a rendszerünk fokozatos hozzáférhetőséget adott a felhasználóknak, egy sikeres támadás esetén csökken az esély arra, hogy a támadó hozzáférjen kritikus rendű információhoz.

Minden egyes csomag tartalmazott egy aláírást és egy egyedi számot. Ha az érkezett információ nem egyezett meg, akkor a rendszer nem engedte tovább a csomagot.

A hálózaton keresztül teljesen lebiztosított adatok mentek végbe az adatok megvédése érdekében.



3. ábra
MAC használata ábrázolva

[5] bemutat és összehasonlít különböző MAC algoritmusokat, figyelmeztetve arra is hogy menynyire nagy veszélyt jelent használni régi algoritmusokat mint például az MD5, mert itt gyakran jönnek létre ütközések és emiatt egyszerűbb feltörni és hozzáférni az aláírásokhoz.

Egy utolsó biztonsági eszközként lehet említeni a hashelést. Hasznos eszköz jelszavak esetén, felhasználtuk a hash-chain változatot biztosabb jelszavak érdekében.

[6] leírja hogyan lehet biztonságosan és helyesen felhasználni hash algoritmusokat a jelszavak bebiztosítása és helyes tárolása érdekében.

3. KÍSÉRLETEK A RENDSZEREN

Az információs csomagokat megfigyeltük Wireshark segítségével, hogy megbizonyosodjunk arról, hogy az üzenetek teljes mértékben titkosítva voltak.

179	7.759933	192.168.1.4	192.168.1.4	TCP	1188 49545 → 12 [PSH, ACK] Seq=13 Ack=13 Win=2619648 Len=1144
91	7.038333	192.168.1.4	192.168.1.4	TCP	676 49523 → 12 [PSH, ACK] Seq=13 Ack=15 Win=2619648 Len=632
191	7.810083	192.168.1.4	192.168.1.4	TCP	240 12 → 49545 [PSH, ACK] Seq=15 Ack=1157 Win=2618624 Len=196
99	7.548536	192.168.1.4	192.168.1.4	TCP	240 12 → 49523 [PSH, ACK] Seq=15 Ack=645 Win=2619136 Len=196

4. ábra
Wireshark észlel csomagokat a hálón

Észrevehető, ahogyan a 4 ábrán Wireshark észlel és kimutat érkező csomagokat. Ez a példa egy üzenet cserét mutat be a kliens és a szerverünk között.

```
02 00 00 00 45 00 00 ec  6d 68 40 00 80 06 00 00   ····E··· mh@·····
c0 a8 01 04 c0 a8 01 04  00 0c c1 73 31 58 09 99   ········ ···s1X··
13 e2 26 5d 50 18 27 f7  8e 2f 00 00 01 00 00 c0   ··&]P·'· ·/·····
39 65 63 38 66 35 66 31  31 61 65 39 39 32 34 37   9ec8f5f1 1ae99247
38 39 33 38 34 32 66 34  35 33 64 39 65 65 62 37   893842f4 53d9eeb7
30 35 33 32 66 30 37 62  37 37 62 65 66 31 38 33   0532f07b 77bef183
64 36 37 35 33 33 66 66  38 32 36 62 32 62 63 63   d67533ff 826b2bcc
38 33 65 62 31 66 35 61  39 33 35 34 62 38 37 34   83eb1f5a 9354b874
38 61 39 36 34 61 38 36  37 32 39 35 64 31 62 33   8a964a86 7295d1b3
64 39 62 66 65 38 36 31  62 35 33 61 61 66 30 34   d9bfe861 b53aaf04
33 65 30 39 64 36 30 36  30 30 62 65 31 63 36 31   3e09d606 00be1c61
38 36 37 36 36 36 33 66  30 63 63 61 61 34 38 65   8676663f 0ccaa48e
31 39 62 63 35 65 39 38  38 30 39 36 34 30 38 37   19bc5e98 80964087
38 33 36 33 38 34 34 62  62 33 33 38 65 61 66 39   8363844b b338eaf9
30 31 61 34 38 37 39 32  62 34 37 30 36 32 65 61   01a48792 b47062ea
```

5. ábra:
Észlelt csomag, aminek a tartalma titkosított

Megfigyelve a 5 ábrát, bővebben látható egy csomagnak a tartalma. Az információ titkosított és a külső szemek nem tudják elolvasni az üzenet tartalmát.

```
945299396543b95870ffdbdfe038813af5e8f0980de355653617d8bde06066ad39eb142b1e026467758e34418884599c2d850d8761fa505f2fb1b63b403e456
31313d2f9785a903e01e89c3fd2b3b32b7575fcdd846c20b97f33e0ef3e776381
GETJOINEDFORUMSLIST#TesterUser#534073b0f3333738a4b157b4bba1825a4140a44bd74d358713657a1bc8ee6815

831a54d9e943bc20b96fdd7b14e972f56c7503093228d5b9d2d4aa4b5930ac5cfd6ccc30f119c1c7432e0033d42f2fe79669450dec43e168d38e47223995df97
f62230e021400e8e26d3ff7837b4a03feae90defaa5a3c0ccc489e977fa832c0fa3c28fd7943907fa4d1e49709a70b65
LATESTFORUMQUESTIONSREQ#Forum1#ad981c71939a632c39e2023b643243604d2488c38d6b40c4a6d4620ffbbaba7e

245628e293d45b75128df600bd54b280cb660f01994c68097261ead2e4ab164345b5f950a61f6adaf1d4df1b3a075958b0d394719a674e678b2c69773038e6aff
711d4c386a9b66b84610676ad3f7693f019d83c4e5b8716336461eca11c3d875f56c78af3cf27241f2a680daa5c9d26082c137bc3110d519f1895fe5c584025
LATESTFORUMQUESTIONSREQ#Forum2#0c23d84f99c7542d60b824b51d63fbbfebe65e77733f57d64ee32fe8f8d783f6

bfeace9a26a240bc9924d33f992c35f029462384358899f43875a164cb4bc9dab8005ebf50d951854930b83f42cda9d408058095484ac78b75a15c7349a28530b3
b4238fb21b3eed75150d88d5cb2122f036c3c47b366ded51ff02599388d7a69d34c43902f0d8b1bc5e9356d6898e45
LATESTFORUMQUESTIONSREQ#Forum3#a5e5cf540870e5299c74e5a322a2b1b9715d50d0d4a72c13c9b3b2846c91fa8c

8f5ec24cb5463f3fe50ec863b62613c3718a51839083e7ca478e45635e39ce999729ef4a78bad32d9a52b7b4cd07c012842be1e3b77510fd5f42f29ca36d898ca66
01ccf57e2bcd6b205370e237c7f3239570d6bc0556103d170b145b8430266
INVITEDFORUMREQ#TesterUser#534073b0f3333738a4b157b4bba1825a4140a44bd74d358713657a1bc8ee6815
```

6. ábra
Szerverhez érkeznek titkosított kérések

Egy újabb példa megmutassa nekünk hogyan néz ki egy átlagos kérése egy kliensnek. A szerverünk várakozik csomagok után, és amint észlel valamit, követvén egy protokollt, elkezdi parciálisan dekodifikálni az üzeneteket.

A 6 ábrán a dekodifikált üzenetek tartalmaznak további, kódolt információs csomagokat. Itt található a kritikusan fontos információ darabok, mint például egy felhasználónak a neve, a jelszava, vagy az üzenete, amit szeretne továbbjuttatni.

Mivel az információk kódolása több kulccsal történik, a támadók nehezebben tudnak hozzáférni a kritikus jellegű információhoz.

```
1#
441293#
84bb04305aba2cbbd215e257775dfee50db18e5535ae46c526ab043f6da7827090fd279bad0ffbca617273f5feca96
f4d4757fcbca641b27085ae9d3c814487d76a269f488600f2f3935628f7172951a6bcb56295dd0b684d938
358358520b43f3cbcbe74d37b037730de90a7938a0f36c33b44de848936b2a4caae7f89f4a98441293#
fc736feb8e1418772f1c37e9db0c7ed2441293#


2#
441293#
264c4a51c0323ee437d21d50e9e0ca957de557e3593390a1cc97cf1785e6d567aa75d525dc496a5cfade30f853a6e0
ee7915cdd4ec133b1208a4ff079fa51f38ff723779e3c723ab314939eff03de23901c3722e8c5d5b8fe39cf
9d652f8edb47cac7febae69924654dc1f71c12ecbf49a770938273d688030da303745f6b770625c826cef7069a4
dbd701574f6a79dc3e6c9e0c71e1457e338860e47ac995d5d37e52a680ce8e72d38359227685dd4b441293#
6a208e20fd58ec694b73bfa3149b56e3441293#


3#
441293#
a871afa0bec68988be90353c7ad36b25ec75ee6dc90aa141c33babe7ca3a96eb7788f2ec83acd66084479c86aab2
ee2bf665cb1bf1f7774c64682db2c56a194e9e2d84db80751b89a8a24a8c4c77958253b411ec6075497a79d1a
a24e881cd5fcd91ed7cb355b659e656512f494b4d2c1799bf2c4a3cfd1fc87576a63800e29f2e560c6252293813f
1438e288db0e33924d66186a323ad415689e7fd2f07c624441293#
51749d2c02da63540542c232eae10302441293#


4#
441293#
f40a23c8ee43dfdbb9cfc171ab261dcbe445d5bbfb85d3db5818003aa620b5df1bae058dba21961c8a2925da76e3
c01b73e0f88b30e9702823f5835b02ea55d20a24fc376b9abd2dff3267648c84bc0d3a59b256f8dbbf153dd975
437ded72c6672cb01022c76124962a02082037fa9dcf5952319d6dd3d7462b1490ed689b59425c45582305ced8983
760482924da9863f206b9c53ccdb70a3d78534d5ad52844fbacb79c530db59d5ebdb0686ba21ad6e04f1c1d53b
b837b2585074261707b441293#
c79bd6197cd7440b77e9be8578f9cbbb


DONE!
```

7. ábra
Titkosított csoport üzenetek

Egy utolsó példaként, a 7 ábránál megfigyelhetjük hogyan néz ki egy beérkező csoportos üzenetezés a szerverünk számára. Fontos itt megemlíteni, hogy a szerverünk, habár képes lenne erre, nem dekodifikálja ezeket az üzeneteket, csak tárolja és továbbítja őket, szükség esetére. Így kizárólag csak a csoporttagok lássák az üzeneteik tartalmát.

KÖVETKEZTETÉSEK

Megfigyelhettük hogyan lehet titkosítani és megnövelni a biztonságát digitális csomagjainknak a multicast felületünknél. Felhasználva a létező, hasznos kriptográfiai eszközöket és biztonsági protokollok alapelveit, kidolgoztunk egy rendszert, ami képes bevédeni, titkosítani, illetve megfigyelni csomagokat és ezeknek a szabályszerű tartalmát, támadások megvédése érdekében.

KÖNYVÉSZET

(1) Rao U.H., N. U. (2014) Cryptography. In: The InfoSec Handbook. Apress, Berkeley, CA. 2014, DOI: https://doi.org/10.1007/978-1-4302-6383-8_8.
(2) Rosler, P.; Mainka, C.; Schwenk, J., More is less: On the end-to-end security of group chats in Signal, WhatsApp, and Threema, 2018, pp 415–429.
(3) Ylonen, T.; Lonvick, C., et al., The secure shell (SSH) protocol architecture; RFC 4251, January: 2006.
(4) Meadows, C. A., Analyzing the Needham-Schroeder public key protocol: A comparison of two approaches, 1996, pp 351–364.
(5) Krawczyk, H.; Bellare, M.; Canetti, R., HMAC: Keyed-hashing for message authentication; RFc 2104, February: 1997.
(6) Security, D. (2019, June 5). Salted Password Hashing. Doing it Right, https://crackstation. net/hashing-security.htm, Accessed: 2021-06-16.