Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

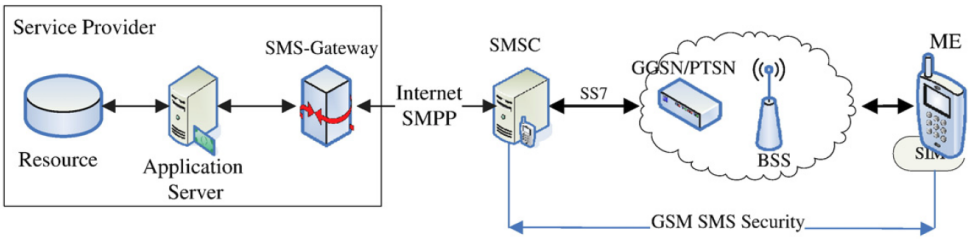
Janez Štular

Varnost sporočil SMS

Projekt pri predmetu
Kriptografija in računalniška varnost

Mentor: prof. dr. Aleksandar Jurišić

Ljubljana, 2017



Slika 1: Pot sporočila SMS

1 Uvod

Storitev SMS (angl. Short Message Service) nam omogoča pošiljanje tekstovnih sporočil preko mobilnega omrežja. Storitev SMS je postala veliko komunikacijsko orodje odkar je bilo 3. decembra 1992 poslano prvo tovrstno sporočilo. Poslal ga je Neil Papworth preko Britanskega mobilnega omrežja Vodafone. Storitev je bila najprej zasnovana kot del globalnega sistema za mobilno komunikacijo GSM (angl. Global System for Mobile Communications), danes pa deluje na širokem naboru omrežnih standardov. Uporaba storitve SMS narašča iz dneva v dan in se uporablja v velikem številu poslovnih aplikacij. Sem spadajo mobilno bančništvo, večnivojska avtentikacija in podobne aplikacije. Storitev SMS je statistično [9] najbolj uporabljena storitev pametnih telefonov, saj jo dnevno uporablja 97% Američanov, ki tako vsak dan pošljejo več kot 6 milijard sporočil SMS. Na svetovnem nivoju se je statistično leta 2015 poslalo kar 8,3 bilijonov tovstnih sporočil, kar je skoraj 23 milijard sporočil dnevno. Poleg tega kar 79% podjetij meni, da stranke želijo imeti podporo sporočil SMS.

Zaradi uporabe sporočil SMS za namene izmenjave zaupnih informacij, poslovnih aplikacij in izmenjave osebnih podatkov, bi bilo potrebno zagotoviti varen prenos sporočil SMS. Zagotovimo ga lahko le z uporabo primernih šifriranih algoritmov za zagotavljanje zaupnosti, integritete podatkov in avtentikacije med dvema končnima uporabnikoma, kar je tudi glavni cilj tega projekta.

Na področju močne avtentikacije so bili že predlagani varnostni mehanizmi uporabe šifriranja z javnimi ključi in sicer v C.S. Park [4] in C.F. Grecas, S.I. Maniatis, I.S. Venieris [5]. Prav tako je bila predlagana generična avtentikacija spletnih storitev z uporabo kartice SIM (angl. Subscriber Identification Module) v A. Wangensteen, L. Lunde, I. Jørstad, T. van Do [6], [7]. Predlagan je bil tudi že protokol SafeSMS od M. Hassinen [8], kateri zagotavlja zaupnost, integriteto ter avtentikacijo in temelji na simetričnem šifriranju in deljeni skrivnosti.

V nadaljevanju sledi opis ranljivosti sporočil SMS in napadov nanje. Preprečimo jih z uporabo predlaganih varnostnih mehanizmov, prvi opisan je protokol SMSSec, ki temelji na Javanski aplikaciji. Opisana je njegova uporaba šifriranih algoritmov, katerim sledi opis poteka prenosa sporočil. Naslednji omenjeni mehanizem je kartica PK-SIM, ki je dodelana kartica SIM. Opisana je njena struktura in uporaba šifriranih algoritmov, predlagana je nova zgradba varnega sporočila SMS in potek njihovega prenosa zaključena s kratko analizo varnosti. Na koncu sledi še zaključek z ugotovitvami in predlogi.

2 Ranljivosti prenosa sporočil SMS

Storitev za pošiljanje sporočil SMS je bila zasnovana za asinhrono delovanje v mobilnih omrežjih. Ta omrežja delujejo ločeno od interneta in so včasih varnejša, manj dostopna in odprta za neželjene namene. Kakorkoli, omrežja GSM ne zagotavljajo pomembnih varnostnih mehanizmov kot so avtentikacija, varnost končnih uporabnikov ali anonimnosti, ki so še posebej pomembni pri poslovnih aplikacijah kot je mobilno plačevanje in bančništvo. Sledi opis nekaterih izmed varnostnih tveganj.

Ranljivosti mobilnih naprav. Mobilne naprave so dovzetne za krajo, zato nam šifriranje celotne poti pošiljanja sporočil SMS ne pomaga, če so dešifrirana sporočila shranjena na nezavarovani mobilni

napravi ali kartici SIM.

Izguba sporočila SMS. Ker so sporočila dostavljena na najboljši možni način (angl. Best Effort Manner), nimamo nobenega zagotovila da bodo prišla do pravega prejemnika. Posledično so lahko sporočila preusmerjena med mobilnimi napravami in strežniki, zaradi česar lahko pride do izgube sporočil. In v primeru, da sporočila nosijo občutljive podatke, to lahko pomeni izgubo zasebnosti.

Ponarejen ID mobilne naprave ali strežnika. Napadalec, ki prestreže spročilo SMS lahko zamenja ID pošiljatelja in se predstavi kot veljavna mobilna naprava ali kot veljaven aplikacijski strežnik. Verjetnost ponarejanja naslovov je zelo velika, saj lahko sporočilo SMS s pravilno nastavljeno glavo pošljemo kar preko interneta. Pri tem pa prejemnik ne more odkriti od kod je bilo sporočilo poslano.

Napad s ponovnim pošiljanjem. Napadalec lahko posname spročilo namenjeno avtentikaciji in ga tako kot pravi pošiljatelj, pošlje prejemniku. S tem se lahko lažno avtenticira in pravemu uporabniku povzroči veliko škode.

Zajem in spreminjanje podatkov med pošiljanjem do bazne postaje. Sporočilo SMS je poslano preko zračnega medija do bazne postaje. Šifriranje omenjene povezave je odvisno od mobilnega operaterja in je v nekaterih državah celo prepovedano. V primeru GSM imamo dva simetrična algoritma za šifriranje; A5/1 s 64 bitnim ključem in A5/2 s 16 bitnim ključem. Oba algoritma sta dovzetna na kriptoanalizo in njune ključe lahko zlomimo v nekaj minutah. Če omrežje ne uporablja šifriranja je običajno označeno z A5/0, pri čemer lahko napadalec neposredno prebere vsebino sporočila. Varnost tu temelji le na veliki količini poslanih sporočil, saj ima napadalec težave z iskanjem zanimivih sporočil SMS.

Ranljivosti poti, ki temelji na protokolu SS7. Varnost sporočil SMS, poslanih preko SS7 omrežja je odvisna od fizične varnosti in uporabe šifrirnih algoritmov, če so le-ti zagotovljeni v teh omrežjih. Njihova izbira je odvisna od mobilnega operaterja in države. Ranljivost so izguba zasebnosti, črpanje občutljivih informacij in napadi DoS (angl. Denial of Service). Omrežje SS7 je fizično varovano s strani večjega števila oseb in ni standardizirano za prenos občutljivih informacij. Protokol GSM MAP je nešifriran in omogoča zaposlenim, ki delajo z omrežjem, da prisluškujejo ali spreminjajo sporočila SMS. Varnost ponovno temelji na velikem številu nezanimivih sporočil.

Ranljivosti centrov za sporočila SMSC. Če je podatkovni center sporočil SMSC (angl Small Message Service Center) dostopen napadalcu, le-ta lahko pregleduje in manipulira sporočila v postopku pošiljanja. Tako kot pri protokolu SS7, tudi na tem nivoju stopnja varnosti ni poznana in se razlikuje med operaterji in državami.

3 Varen prenos sporočil s protokolom SMSSec

Protokol SMSSec ima dve glavni omejitvi s katerima se sooča:

- omejena procesorska moč in količina sistemskega pomnilnika nekaterih mobilnih naprav, za zagotavljanje zasebnosti in integritete sporočil SMS, mora protokol uporabiti šifrirne algoritme, ki so lahko računsko zelo zahtevni, posebaj velja omeniti tiste za delo z javnimi ključi,
- struktura in dolžina sporočil SMS, mora biti v skladu z vsemi tipi sporočil SMS, zato mora šifriranje zagotoviti naslednje tri lastnosti:
  - šifrirano sporočilo SMS mora ustrezati telesu klasičnega sporočila SMS,
  - šifriranje mora ohranjati dolžino sporočila SMS, saj bi v primeru dolgih sporočil presegli največjo dovoljeno dolžino,
  - šifriranje mora biti enostavno in računsko nezahtevno.

3.1 Uporaba šifrirnih algoritmov

Za zagotavljanje varnosti je potrebno uporabiti tako simetrično kot tudi asimetrično šifriranje in MAC (angl. Message Authentication Code) integriteto sporočil SMS. Zagotoviti je potrebno varnost in učinkovitost protokola, pri čemer ima varnost večji pomen. Posledično je potrebno biti previden z izbiro varnih šifrirnih algoritmov. Protokol SMSSec za zagotavljanje varnosti predpisuje naslednje:

- simetrično šifriranje AES (angl. Advanced Encryption Standard) zaradi njegove hitrosti in standardizacije v načinu delovanja CTR (angl. Counter Mode), ki je enostaven za implementacijo, pozornost je potrebno posvetiti le števcu, ki mora biti edinstven za vsak ključ, pravi 128 bitni sistem ima priporočljivo dolžina ključa 256 bitov, saj je šifrirano sporočilo lahko tarča rojstnodnevnega napada,

- asimetrično šifriranje RSAES (angl. Ron Rivest Adi Shamir Encryption Scheme) v kombinaciji z OAEP (angl. Optimal Asymmetric Encryption Padding), šifriranje RSA je že dolga leta deležno zaupanja in še vedno ostaja kot šifrirni algoritem poslovnih aplikacij, funkcija OAEP je vključena z razlogom, da pokvari matematično strukturo sporočil, ki so lahko del mobilnih aplikacij, zaradi problema faktorizacije velikih števil je za naslednjih 10 let predlagana dolžina ključa 2048 bitov, načrtovalci protokola niso želeli uporabiti šifriranja na eliptičnih krivuljah kljub njihovi večji učinkovitosti in krajši dolžini ključa, navedeni razlog je veliko število patentov posameznikov in podjetij po celotnem svetu, Kanadsko podjetje Certicom Inc. naj bi pod svojim okriljem imelo več kot 130 patentov šifriranja na eliptičnih krivuljah,

- integriteta sporočil HMAC (angl. Keyed-Hash Message Authentication Code) zagotovljena s funkcijo za digitalne izvlečke SHA256 (angl. Secure Hash Algorithm), zaradi napadov in kolizij algoritem SHA1, z integriteto HMAC zagotovimo zaščito sporočil pred delnimi kolizijami in napadom podaljšane dolžine (angl. Length Extension Attack).

3.2 Potek prenosa

Protokol SMSSec uporablja asimetrično in simetrično šifriranje sporočil ter dvo-stopenjsko avtentikacijo. Vsebuje tri kriterije za avtentikacijo uporabnika: nekaj kar uporabnik ve, npr. geslo; nekaj kar uporabnik ima, npr. pametno kartico in nekaj kar uporabnik je, npr. odtis njegovega palca. Dvostopenjska avtentikacija je katerakoli kombinacija, ki zahteva uporabo dveh izmed zgornjih kriterijev.

Na tej stopnji postane vprašljiva uporaba šifriranja RSA, saj njegova uporaba preseže velikost 140 bajtov pri uporabi 2048 bitnega ključa. To pomeni, da za prenos šifrirnega sporočila potrebujemo več kot eno sporočilo SMS. Za zagotovitev večje učinkovitosti je protokol SMSSec razdeljen na dva ločena poteka rokovanj, prvo in naslednje.

Prvo rokovanje katerega potek je opisan v nadaljevanju.

M1: $C_{p_z} \rightarrow S_{p_z}\ \{U||H||D||Q||R_c\}_{PK}$

Za začetek rokovanja uporabnik vnese geslo *PIN* in svojo telefonsko številko *U*. Pri tem geslo *PIN* predstavlja nekaj, kar uporabnik ve. Geslo *PIN* je znano obema, uporabniku in strežniku. Poleg tega telefonska številka predstavlja nekaj, kar uporabnik ima.

Mobilni telefon nato generira naslednje stvari: parametre za generiranje simetričnega ključa *D*, nov identifikator seje *Q*, HMAC (preko telefonske številke, gesla *PIN* in indentifikatorja seje), *H* ter naključen izziv R_c. S tem zagotovi svež potek protokola.

Najlažje in verjetno najučinkoviteje je, da *D* vsebuje psevdo naključno 256 bitno sol, ki je po možnosti tudi število ponovitev *r* (angl. round) uporabljenih za razpihovanje gesla *PIN*. Z uporabo algoritmov za digitalne izvlečke lahko mobilna naprava izračuna simetrični ključ *SK* kot:

$X_0 := 0$

$X_i :=$ izvleček(X_{i-1},*PIN*, *sol*); za i = 1,...,r

SK:= X_r

Parameter *r* naj bo čim večji. Priporočena funkcija za izvleček je SHA256, parameter *r* pa naj bo izbran tako, da računanje *SK* traja od 200 do 1000 ms in naj nikoli ne bo fiksno izbran. Pomembno je, da lahko tako mobilna naprava, kot tudi strežnik izračunata simetrični ključ *SK* iz gesla *PIN*. Simetrični ključ *SK* ni med rokovanjem nikoli prenešen. Skrivnost za računanje izvlečka je lahko *sol*, pridobljena iz parametrov za vzpostavitev simetričnega ključa *D*.

Ker je pošiljanje sporočil SMS definirano s stanji, lahko identifikator seje *Q* uporabimo za sledenje komunikaciji. Na samem začetku je vrednost *Q* na mobilni napravi 1 in na strežniku 0, saj mora biti *Q* na mobilni napravi vedno večji kot na strežniku, da s tem preprečimo napad s ponovnim pošiljanjem. Pri računanju HAMC *H* je pomemben vrstni red elementov, poleg tega njihova skupna velikost ne sme presegati 190 bajtov zaradi 256 bajtnega kodiranja OAEP.

Celotno sporočilo M1 je zašifrirano s strežnikovim javnim ključem *PK*. Pred pošiljanjem se tako na mobilno napravo shranita tudi vrednosti *D* in *Q*. Sporočilo je poslano preko vnaprej definiranih vrat protokola P_z.

Ko strežnik prejme sporočilo M1 ga dešifrira s svojim zasebnim ključem. Če se sporočilo uspešno dešifrira, strežnik najprej v svoji bazi preveri geslo *PIN*, ki ga je poslal uporabnik. Nato preveri izvleček *H* in identifikator seje *Q* ter s tem zagotovi integriteto sporočila in prepreči napad s ponovnim pošiljanjem. Če se izračunani izvleček ne ujema s prejetim ali vrednost *Q* ni enaka 0, strežnik zavrže sporočilo in zaključi protokol.

Preden stežnik odgovori s sporočilom M2, v svojo varno bazo shrani izvleček telefonske številke *HU*, parametre za pridobitev ključa *D* in identifikator seje *Q*.

M2: $S_{p_z} \rightarrow C_{p_z}$ $\{R_c||P_j||SQ\}_{SK}$

Ko je uporabnik avtenticiran, strežnik odpre nova vrata za pogovor P_j, ki so namenjena nadaljni komunikaciji. Za vsakega uporabnika so odprta nova vrata, zato mora protokol skrbeti za njihovo vodenje. Namenjena so večji skalabilnosti in zasebnosti. Zaporedna številka *SQ* se začne pri 1 in označuje prvo sporočilo zašifrirano s simetričnim algoritmom. Strežnik zašifrira sporočilo M2 s pomočjo simetričnega ključa *SK*, ki ga generira na podoben način kot uporabnik pri sporočilu M1.

Zaradi uporabe simetričnega šifriranja lahko strežnik pošlje vsebino znotraj enega sporočila SMS in prihrani tudi na računanju. Ko uporabnik sprejme sporočilo, ga dešifrira in preveri naključni izziv R_c. Strežnik je avtenticiran, če uporabnik uspešno dešifrira sporočilo in preveri izziv,ker le pravi strežnik lahko zgenerira simetrični ključ *SK*.

Če uporabnik ne more preveriti izziva, zavrže sporočilo M2 in konča s protokolom, če pa uspešno preveri sporočilo se rokovanje zaključi.

M3: $C_{p_z} \rightarrow S_{p_j}$ $\{M||SQ\}_{SK}$

M4: $S_{p_j} \rightarrow C_{p_z}$ $\{M||SQ\}_{SK}$

M3 in M4 sta podsekvenci sporočil SMS med uporabnikom in strežnikom. Strežnik vedno pošilja sporočila na vrata P_z, uporabnik od sedaj naprej na vrata P_j. Šifriranje celotne komunikacije je simetrično in uporablja ključ *SK*. Obenem morata tako uporabnik kot tudi strežnik skrbeti za povečevanje vrednosti zaporedne številke *SQ*. S tem varujeta sporočila pred napadom s ponovnim pošiljanjem. V primeru neustrezne številke *SQ* ali pretečenega prednastavljenega časa na strani uporabnika ali strežnika, si tega med seboj ne potrebujeta posredovati, ampak morata le zaključiti s protokolom.

Naslednja rokovanja se razlikujejo, saj ni učinkovito vedno uporabljati RSA šifriranje tekom rokovanja, ker le-to terja dve poslani sporočili M1. Poleg tega simetrično šifriranje porabi tudi manj računske moči, zato se naslednja rokovanja razlikujejo od prvega in imajo sledeč potek:

M1: $C_{p_z} \rightarrow S_{p_z}\ \{U||H||D||Q||R_c\}_{SK}$

Generiranje sporočila M1 je podobno kot pri prvem rokovanju, le da tu uporabljamo simetrično šifriranje s ključem *SK* namesto asimetričnega. Zaradi identifikacije uprabnika, le-ta na začetku sporočila pripne izvleček telefonske številke. Generirati je potrebno nove parametre za generiranje novega simetričnega ključa D_n in nov identifikator seje Q_n, ki je le za ena povečan *Q*. Celotno sporočilo je zašifrirano s prejšnjim simetričnim ključem *SK*. Preden uporabnik pošlje sporočilo M1, zamenja stari vrednosti *D* in *Q* z novima D_n in Q_n.

Ko strežnik prejme sporočilo M1, ga dešifrira in preveri izvleček telefonske številke *HU* ter ga primerja s tistim iz prejšnjega rokovanja. Če se izvleček *HU* ne ujema s prejšnjim, strežnik zaključi s protokolom. V nasprotnem primeru pa uporabi stare parametre za simetrični ključ *D* in s pomočjo starega simetričnega ključa *SK* dešifrira sporočilo. Strežnik ga dešifrira in podobno kot pri prvem rokovanju preveri vrednosti *H* in *Q* z zahtevo, da je nova vrednost *Q* večja od stare vrednosti *Q*. Uporabnik je tako avtenticiran po preverbi vrednosti *H* in *Q*, ker je zašifriral sporočilo M1 z istim simetričnim ključem *SK*, kot ga je zgeneriral pri prejšnjem rokovanju. Preden strežnik pošlje sporočilo M2, zamenja stare vrednosti *D*, *Q* in *HU* z novimi D_n, Q_n in HU_n.

M2: $S_{p_z} \rightarrow C_{p_z}\ \{R_c||P_j||SQ\}_{SK_n}$

Ko je uporabnik avtenticiran, strežnik zašifrira sporočilo M2 enako kot sporočilo pri prvem rokovanju, le da tukaj uporabi simetrični ključ SK_n, ki je generiran enako kot pri prvem rokovanju z uporabo novega D_n.

Ko uporabnik prejme M2 uporabi svoj generiran simetrični ključ SK_n in dešifrira sporočilo. Strežnik je avtenticiran, če uporabnik lahko uspešno dešifrira sporočilo in uspešno preveri naključni izziv Rc, ker le pravi strežnik lahko generira SK_n. V nasprotnem primeru uporabnik zaključi s protokolom. Po preverbi strežnika je rokovanje zaključeno.

M3: $C_{p_z} \rightarrow S_{p_j}\ \{M||SQ\}_{SK_n}$

M4: $S_{p_j} \rightarrow C_{p_z}\ \{M||SQ\}_{SK_n}$

Podsekvenca sporočil M3 in M4 je enaka kot pri prvem rokovanju. Razlikuje se le v tem, da so sporočila zašifrirana s simetričnim ključem SK_n.

4 Varen prenos sporočil s kartico PK-SIM

Predlagana je nova kartica PK-SIM, ki je podobna kartici SIM (angl. Subscriber Identity Module) z dodatno vgrajenimi komponentami za podporo infrastrukturi javnih ključev PKI (angl. Public key infrastructure). Kartica PK-SIM je zaščitena proti nepooblaščenim posegom (angl. tamper resistant) in zaščitena z geslom PIN kot običajna kartica SIM. Glavni cilj kartice PK-SIM je nadgradnja obstoječe kartice SIM, ki bo lahko hranila skrivne ključe. Poleg tega bo hranila in izvajala tudi šifrirane algoritme na svojem čipu z namenom, da ne bo izpostavljala skrivnih ključev drugim napravam.

4.1 Struktura kartice PK-SIM in uporaba šifrirnih algoritmov

Uporabniki GSM storitev so v omrežje prijavljeni preko naprave, varne pred posegi imenovane kartica SIM, ki je klasična pametna kartica z GSM aplikacijami. Temelji na pametni kartici s procesorjem in je uporabljena za hrambo kode ID in skrivnega ključa uporabnika. Prav tako omogoča avtentikacijo

uporabnika v omrežje GSM. Tudi kartica PK-SIM ima enake lastnosti kot klasična SIM in se lahko uporablja v katerikoli mobilni napravi. Poleg tega ima vgrajen pomožni procesor, pomnilnik, pomnilni prostor za hrambo zasebnih ključev in nekatere komponente za delo z javnimi ključi PKI. Razširitve omenjene kartice so za operaterja mobilne telefonije povsem transparentne. Glavna razlika je v tem, da je varno sporočilo hranjeno direktno na kartici PK-SIM, ki vsebuje naslednje logične enote:

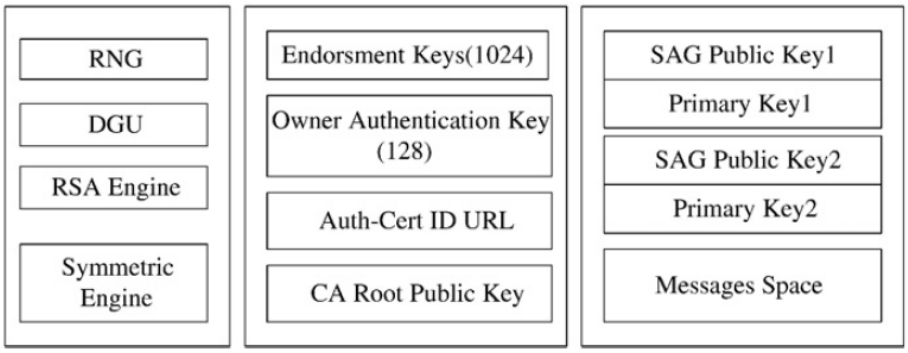
- generator psevdo naključnih števil RNG, ki je strojno podprt in generira kvalitetna naključna števila,
- enoto za generiranje digitalnih izvlečkov DGU, podprto z algoritmom MD5 kateri proizvaja izvlečeke HMAC,
- enoto RSA, ki je strojno podprta in lahko dela z 1024 bitov velikimi ključi, ki temeljijo na naključnih številih enote RNG, namenjena je podpisovanju, preverjanju podpisov, šifriranju in dešifriranju,
- enoto za simetrično šifriranje, ki podpira algoritma 3DES (angl. Data Encryption Standard) in AES.

Enote za izvlečke, RSA in simetrično šifriranje so implementirane v zbirnem jeziku in se izvajajo na šifrirnem pomožnem procesorju kartice. Poleg tega imajo za svoje izvajanje ločen pomnilni prostor. Kartica PK-SIM hrani ključe in sporočila, kot je opisona v nadaljevanju:

- RSA 1024 bitni par zasebnega in javnega ključa, ki sta naključno generirana na kartici z modulom RSA, ko je kartica izdana uporabniku in poleg tega ne moreta biti zamenjana, zasebni ključ je shranjen v posebej varni datotečni strukturi in nikoli ne zapusti kartice PK-SIM, s čimer je preprečeno kakršnokoli vohunjenje za ključi, posledično mora biti dešifriranje in podpisovanje izvedeno na kartici, javni ključ je shranjen v certifikatu mobilne naprave, uporablja se za potrjevanje in šifriranje občutljivih podatkov, ki so poslani kartici, ključa sta občutljiva z vidika zasebnosti, tako da ju lahko uporabnik popolnoma onemogoči,
- zaradi omejenega prostora kartice, ta hrani le naslov URL (angl. Uniform Resource Locator) do svojega javnega ključa oziroma certifikata naprave, podpis katerega lahko preveri z javnim ključem certifikatne agencije CA, ki je na začetku prazen in dobi vrednost ob izdaji kartice ter je kasneje nikoli ne zapusti,
- skrivni 128 bitni avtentikacijski ključ vezan na geslo PIN, uporabljen za avtentikacijo občutljivih uporabniških ukazov, kot npr. uporaba privatnega ključa ali branje varnih sporočil SMS,
- kartica PK-SIM ima dve enoti za hranjenje javnega ključa, primarno deljenega ključa in menijsko vodene aplikacije, mape s temi enotami so lahko odstranjene dinamično, posledično pa je uporabnik lahko prijavljen na dve omrežji hkrati,
- kartica ima poleg omenjenega tudi prostor za varna sporočila SMS, ki jih je možno prebirati le z vnosom gesla PIN, ko je prostor zapolnjen, uporabnik ne more več prejemati sporočil, presežek se hrani v omrežju in ga uporabnik lahko prejme, ko sprosti prostor na kartici.

4.2 Zgradba varnega sporočila SMS

Sporočilo SMS je sestavljeno iz dveh delov in sicer iz varnostne glave in šifriranih uporabniških podatkov. Varnostno glavo sestavlja 12 bajtov, ki opisujejo namen in varovanje podatkov sporočila. Celotno sporočilo sestavlja 140 bajtov, tako da nam za šifrirane podatke ostane 128 bajtov, kar pomeni 91,4% učinkovitosti prenosa.



Slika 2: Logična stroktura kartice PK-SIM

- zastavica 1 bajt služi kot indikator varnosti in storitev, če je prvi bit enak 1; gre za varno sporočilo, naslednjih 7 bitov določa kodno ime ciljne aplikacije,
- stil aplikacije 1 bajt, katerega prvi 3 biti predstavljajo verzijo protokola in naslednjih 5 bitov stil sporočila, pošiljatelj in prejemnik morata uporabljati enako verzijo protokola, ki vsebuje 7 različnih sporočil za avtentikacijo, poganja za primarni ključ, podatkovni prenos in nekaj statusnih sporočil,
- algoritem 1 bajt, ki identificira uporabljeni šifrirni algoritem, zaradi prostorske omejitve imamo tu le dva simetrična algoritma 3DES in AES,
- števec 1 bajt, kateri šteje vsa prenešena sporočila in katerega naloga je preprečevanje napadov s ponovnim pošiljanjem, pred pošiljanjem obe strani povečata števec za 1, če je prejeti števec večji od trenutnega ga prejemnik sprejme, v nasprotnem primeru ga zavrže,
- faktor diverzifikacije 4 bajte, ki je naključno generirana vrednost *Na* poslana kot čistopis, uporablja se za generiranje ključa in seje s pomočjo širitve primarnega ključa,
- MAC 4 bajte, prvi 4 bajti digitalnega izvlečka (*Na*, primarnega ključa, sporočila) služijo integriteti podatkov,
- zašifrirani podatki 128 bajtov, zašifrirani z algoritmom, ki uporabljajo ključ generirni s pomočjo vrednosti zgornjega faktorja diverzifikacije *Na*.

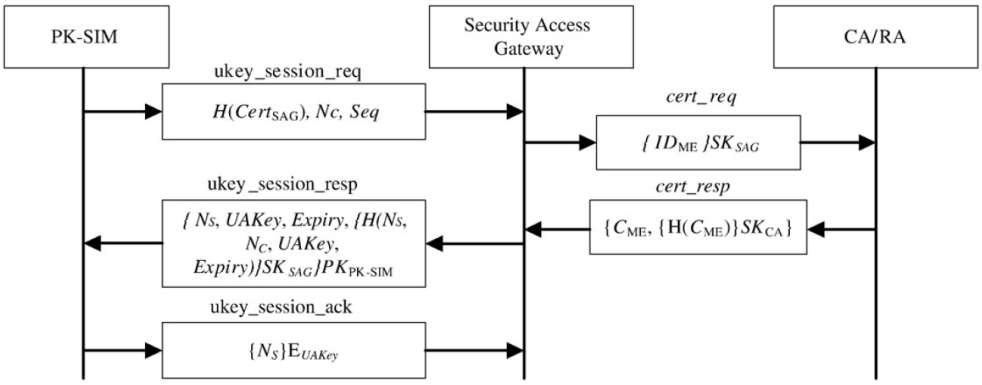
4.3 Potek prenosa

Prenos je sestavljen iz dveh faz: faze avtentikacije ter faze vzpostavitve sejnega ključa in komunikacije. Faza avtentikacije omogoča kartici PK-SIM in varnemu prehodu SAG (angl. Security Access Gateway), da avtenticirata drug drugega z uporabo javnih ključev in vzpostavitvijo primarne deljene skrivnosti. Prva faza omogoča kartici PK-SIM in prehodu SAG kreiranje simetričnega sejnega ključa za šifriranje, dešifriranje ter odkrivanju posegov v sejo. Najdaljšo veljavnost ključa določi prehod SAG in je lahko osvežena avtomatično ali ročno. Sejni ključ je veljaven le za eno transakcijo, zato mora biti za vsako naslednjo generiran nov.

Faza avtentikacije, katere opis sledi v nadaljevanju.

M1: PK-SIM $\rightarrow$ SAG $UAzahteva_{seja} = \{H(cert_{SAG}), ID_M, N_c, Seq\}$

V fazi avtentikacije se uporabnik s pomočjo certifikata avtenticira prehodu SAG ter pridobi primarni ključ. Da poenostavimo, bomo predpostavili, da je certifikat naložen na kartico PK-SIM. Faza avtentikacije se prične z vzpostavitvijo primarnega ključa. Kartica generira dve naključni števili N_c



Slika 3: Faza avtentikacije protokola PK-SIM

in *Seq* ter izračuna digitalni izvleček *H* certifikata vrat SAG. Vse to in naslov do svojega certifikata ID_M pošlje vratom SAG.

M2: SAG $\rightarrow$ CA $zahteva_{certifikat} = \{ID_M\}_{SK_{SAG}}$

Vrata SAG prejmejo sporočilo M1 $UAzahteva_{seja}$, preveri izvleček *H*, če se ujema z njegovim odtisom (angl. Fingerprint). Če je preverba neuspešna, mora kartica PK-SIM najprej pridobiti pravi javni ključ prehoda SAG. V nasprotnem primeru pa prehod SAG iz sporočila prebere povezavo do certifikata mobilne naprave ID_M in preveri njegovo veljavnost. Nato pošlje $zahteva_{certifikat}$ certifikatni agenciji CA. To sporočilo je sestavljeno iz povezave do certifikata mobilne naprave ID_M in je podpisano z zasebnim ključem SK_{SAG} prehoda SAG:

M3: CA $\rightarrow$ SAG $odgovor_{certifikat} = \{C_M, \{H(C_m)\}_{SK_{CA}}\}$

Certifikatna agencija CA prejme $zahteva_{certifikat}$ in preveri povezavo do certifikata mobilne naprave ID_M na strežniku. Če pod tem naslovom najde veljaven certifikat, ga pošlje prehodu SAG. V nasprotnem primeru prehodu SAG sporoči napako. Sporočilo $odgovor_{certifikat}$ je tako sestavljen iz certifikata mobilne naprave C_M in digitalnega izvlečka omenjenega certifikata podpisanega z zasebnim ključem certifikatne agencije $\{H(C_m)\}_{SK_{CA}}$.

M4: SAG $\rightarrow$ PK-SIM

$UAodgovor_{seja} = \{N_S, UAkey, expirity, \{H(N_S, N_C, UAkey, ID_M, expirity)\}_{SK_{SAG}}\}_{PK_M}$

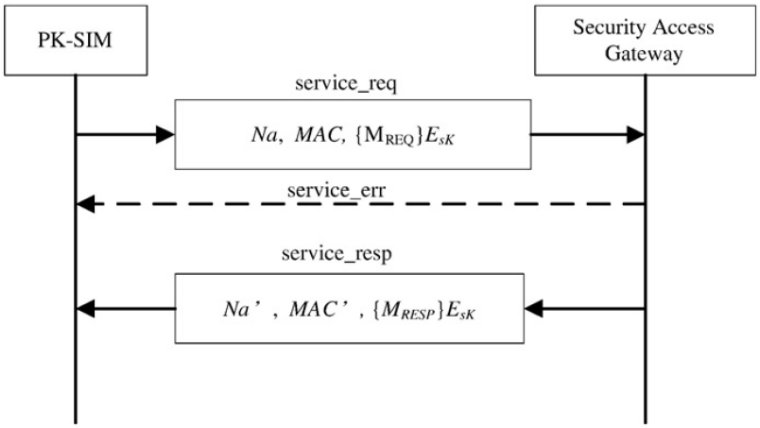
Ko prehod SAG prejme sporočilo $odgovor_{certifikat}$, zgenerira naključno število N_S in izračuna privzeti ključ *UAkey* s pomočjo funkcije, ki jo kot argument poda $N_C||N_S$. Za zagotovitev integritete nato prehod SAG izračuna še izvleček $H(N_S, N_C, UAkey, ID_m, expirity)$, v katerega vključi tudi čas poteka *expirity* ključa *UAkey*. Tako sestavi sporočilo $UAodgovor_{seja}$, ki ga zašifrira z javnim ključem $PK_{PK\text{-}SIM}$ in ga pošlje kartici PK-SIM.

M5: PK-SIM $\rightarrow$ SAG $UApotrditev_{seja} = \{N_s\}_{E_{UAkey}}$

Ko kartica prejme sporočilo, ga dešifrira in preveri ustreznost podatkov s ponovnim računanjem digitalnega izvlečka $H(N_S, N_C, UAkey, ID_M, expiry)$, z uporabo svojega števila N_c in povezavo do certifikata mobilne naprave ID_m. Če se izvleček ujema, kartica sprejme privzeti ključ *UAkey* in čas veljavnosti *expirity*. Nato ustreznost ključa potrdi prehodu SAG s sporočilom $seja_{potrditev}$, katero vsebuje naključno število N_S prehoda SAG in je zašifrirano s primarnim ključem *UAkey*.

Ko prehod SAG prejme sporočilo in ga uspešno dekriptira je privzeti ključ $UAKey$ vzpostavljen.

Vzpostavitev primarnega ključa *UAKey* je najdaljši del protokola, zato se običajno izvaja enkrat dnevno ali na daljše časovno obdobje.



Slika 4: Faza vzpostavitve sejnega ključa in komunikacije protokola PK-SIM

Faza vzpostavitve sejnega ključa in komunikacije, če uporabnik želi varno poslati sporočilo SMS do prehoda SAG, mora preveriti veljavnost primarnega ključa *UAkey* in pričeti s fazo vzpostavitve sejnega ključa, ki temelji na primarnem ključu in je opisana v nadaljevanju.

M1: PK-SIM $\rightarrow$ SAG $zahteva_{storitev} = \{Na, ID_M, \{m\}_{E_{K_s}}, H\{Na||UAkey||m\}\}$

Kartica PK-SIM generira naključno število *Na* dolgo 4 bajte imenovano faktor diverzifikacije in s pomočjo funkcije ki ji poda (*UAKey,Na*), izračuna sejni ključ K_s. Nato zašifrira željeno sporočilo *m* z omenjenim sejnim ključem. Pred njim pripne faktor *Na* in povezavo do certifikata mobilne naprave ID_M ter zatem vse skupaj pripravi za pošiljanje prehodu SAG. Za preprečitev nepooblaščenega posega v sporočilo se čez prej omenjeni del izračuna MAC.

M2: SAG $\rightarrow$ PK-SIM $odgovor_{storitev} = \{Na, ID_M, \{m\}_{E_{K_s}}, H\{Na||UAkey||m\}\}$

Ko prehod SAG prejme spročilo $zahteva_{storitev}$ iz njega prebere ID_m in *Na*. Za prejeti naslov mobilne naprave poišče pripadajoči *UAkey* ter z njegovo pomočjo izračuna sejni ključ K_s. S pomočjo izračunanega sejnega ključa odšifrira sporočilo *m* ter izračuna izvleček MAC in ga primerja s prejetim. Če se izvlečka MAC ujemata, v sporočilo ni bilo posegano. V nasprotnem primeru prehod SAG sporočilo zavrne in kartici PK-SIM pošlje sporočilo o napaki, da mu ta lahko ponovno pošlje sporočilo. Pošiljanje varnega sporočila iz prehoda SAG na kartico PK-SIM je enako kot v prvem koraku.

V tej fazi se prehod SAG in kartica PK-SIM avtenticirata eden drugemu na podlagi primarnega ključa *UAkey* s pomočjo katerega generirata sejni ključ K_s. Razvidno je, da je sejni ključ uporabljen le enkrat (angl. One-Time Pad Key) in s tem je preprečen napad s ponovnim pošiljanjem. Če poteče primarni ključ *UAkey*, prehod SAG zavrne sporočilo in obvesti kartico PK-SIM da bo potrebna ponovna vzpostavitev primarnega ključa *UAkey*. V tem protokolu je veljavnost primarnega ključa določena s pravili prehoda SAG. Običajno je primarni ključ *UAkey* lahko uporabljen za več 100 sporočil, saj lomljenje 1024 bitnega sejnega ključa K_s s tradicionalnimi metodami traja nekaj 10 let.

4.4 Analiza varnosti

Prva in zelo pomembna je hramba sporočil SMS, kar v primeru uporabe kartice PK-SIM ni nobena zahtevna naloga. Sporočila se hranijo na sami kartici, ki je odporna proti fizičnim nepooblaščenim posegom (angl. tamper resistant) in zaščitena z geslom PIN. Če želimo priti do sporočil SMS moramo izvesti nepooblaščeni fizični poseg ali zlomiti geslo PIN.

Fizična varnost pametnih kartic je trenutno zagotovljena in je temelj za naprednejšo kontrolo dostopa, bančne kartice in podobno. Posledično lahko sklepamo, da so sporočila na kartici PK-SIM s fizičnega vidika varna, dokler so varne pametne kartice.

Druga možnost je lomljenje gesla PIN, ki je v najslabšem primeru sestavljeno iz štirih števk in ga lahko trikrat poskusimo ugotoviti. Za vsako števko imamo na voljo 10 možnosti, kar pomeni, da lahko sestavimo $10 \cdot 10 \cdot 10 \cdot 10 = 10000$ gesel PIN. Verjetnost uspešnega lomljenja gesla PIN je enaka seštevku treh verjetnosti: verjetnost, da geslo uganemo v prvem poizkusu; verjetnost, da geslo uganemo v drugem poizkusu; verjetnost, da geslo uganemo v tretjem poizkusu. Ti trije dogodki so paroma nezdružljivi, zato je njihova verjetnost enaka vsoti verjetnosti teh treh dogodkov in jo lahko zapišemo z enačbo:

$P(A) = P(A_1 + A_2 + A_3) = P(A_1) + P(A_2) + P(A_3)$

Začnimo s prvim poizkusom, dogodek da uganemo geslo PIN naj bo U, pri čemer je njegova verjetnost enaka $P(U) = \frac{1}{10000}$. Posledično je verjetnost da geslo uganemo v prvem poizkusu enaka verjetnosti tega dogodka:
$P(A_1) = P(U) = \frac{1}{10000}$

Če gesla PIN v prvem poizkusu ne uganemo nadaljujemo s poizkušanjem oziroma dogodkom A_2, pri čemer upoštevamo enačbo za poljubna dva dogodka,
$P(\overline{A}_1 \cup A_2) = P(\overline{A}_1) + P(A_2) - P(\overline{A}_1 \cap A_2)$
$P(A_2) = P(\overline{A}_1 \cup A_2) - P(\overline{A}_1) + P(\overline{A}_1 \cap A_2)$, pri čemer upoštevamo, da je dogodek A_2 enak dogodku U, saj v tem primeru geslo uganemo in dogodek $\overline{A}_1$ je enak dogodku $\overline{U}$ katerega verjetnost je $P(\overline{U}) = 1 - P(U) = 1 - \frac{1}{10000} = \frac{9999}{10000}$, kjer gesla ne uganemo. Iz tega sledi:
$P(A_2) = P(\overline{U} \cup U) - P(\overline{U}) + P(\overline{U} \cap U) = 1 - \frac{9999}{10000} + 0 = \frac{1}{10000}$

Če gesla PIN še vedno nismo uganili lahko poizkusimo z zadnjim ponovnim ugibanjem in upoštevamo enačbo za poljubne tri dogodke:
$P(\overline{A}_1 \cup \overline{A}_2 \cup A_3) = P(\overline{A}_1) + P(\overline{A}_2) + P(A_3) - P(\overline{A}_1 \cap \overline{A}_2) - P(\overline{A}_1 \cap A_3) - P(\overline{A}_2 \cap A_3) + P(\overline{A}_1 \cap \overline{A}_2 \cap A_3)$
$P(A_3) = P(\overline{A}_1 \cup \overline{A}_2 \cup A_3) - P(\overline{A}_1) - P(\overline{A}_2) + P(\overline{A}_1 \cap \overline{A}_2) + P(\overline{A}_1 \cap A_3) + P(\overline{A}_2 \cap A_3) - P(\overline{A}_1 \cap \overline{A}_2 \cap A_3)$,
kjer ponovno lahko upoštevamo da sta dogodka $\overline{A}_1$ in $\overline{A}_2$ enaka dogodku $\overline{U}$ in da je dogodek A_3 enak dogodku U iz česar sledi:
$P(A_3) = P(\overline{U} \cup \overline{U} \cup U) - P(\overline{U}) - P(\overline{U}) + P(\overline{U} \cap \overline{U}) + P(\overline{U} \cap U) + P(\overline{U} \cap U) - P(\overline{U} \cap \overline{U} \cap U) =$
$= P(\overline{U} \cup U) - P(\overline{U}) - P(\overline{U}) + P(\overline{U}) + P(\overline{U} \cap U) + P(\overline{U} \cap U) - P(\overline{U} \cap U) =$
$= P(\overline{U} \cup U) - P(\overline{U}) + P(\overline{U} \cap U) = 1 - \frac{9999}{10000} + 0 = \frac{1}{10000}$

Z vsem tem smo dokazali dve stvari in sicer:
$P(U) = P(A_1) = P(A_2) = P(A_3) = \frac{1}{10000}$
$P(A) = 3 \cdot \frac{1}{10^4} = \frac{3}{10^4}$

Geslo PIN tako lahko ugotovimo z verjetnostjo $3 \cdot 10^{-4}$, pri tem moramo upoštevati, da je verjetnost uganitve gesla PIN fiksna in ga po pretečenih treh poizkusih ne moremo več ugibati. Kartica PK-SIM se nato zaščiti z geslom PUK, ki je sestavljeno iz osmih števk in ga ponovno lahko poizkusimo ugotoviti, saj je njegova verjetnost ugotovitve enaka seštevku neodvisnih dogodkov oziroma verjetnosti, tako kot pri geslu PIN in sicer:

$P(B) = P(B_1 + B_2 + B_3) = P(B_1) + P(B_2) + P(B_3)$

Ponovno potrebujemo dogodek, ko smo ugotovili geslo U, katerega verjetnost je $P(U) = \frac{1}{10^8}$, ker imamo 8 števk z 10 različnimi možnostmi. Odvisnost med dogodki je enaka kot pri geslu PIN, zato lahko uporabimo prej dokazani dejstvi:
$P(V) = P(B_1) = P(B_2) = P(B_3) = \frac{1}{10^8}$
$P(B) = 3 \cdot \frac{1}{10^8} = \frac{3}{10^8}$

Ko zaključimo s poizkušanjem gesla PUK smo opravili največ 3 poizkuse. Če ga v tem času nismo uspeli ugotoviti smo kartico PK-SIM zaklenili in zaključili s poizkušanjem. Dogodka, da smo ugotovili geslo PIN A in da smo ugotovili geslo PUK B sta medsebojno nezdružljiva, zato lahko verjetnost ugotovitve gesla izračunamo kot seštevek njunih verjetnosti, pri čemer C predstavlja dogodek, da smo geslo ugotovili:

$P(C) = P(A) + P(B) = \frac{3}{10^4} + \frac{3}{10^8} = \frac{3 \cdot (10^4+1)}{10^8} \approx \frac{3}{10^4} < \frac{4}{10^4}$

Iz tega sledi, da je najšibkejši del varnosti hranjenja sporočil moč gesel PIN in PUK. V najslabšem primeru je verjetnost ugotovitve gesla fiksna in je približno 3 ‰, kar pomeni, da je hramba sporočil na kartici PK-SIM varna.

Varnostni sistem je varen toliko, kolikor je varen njegov najšibkejši člen. Če predpostavimo, da so uporabljeni šifrirani algoritmi semantično varni, kar pomeni, da za njihovo lomljenje potrebujemo približno 2^{80} poizkusov oziroma je verjetnost ugotovitve vsebine šifriranega sporočila enaka 2^{-80}, potem je v našem primeru najšibkejši člen geslo PIN oziroma PUK. Iz tega sledi, da je uporaba kartice PK-SIM varna in je verjetnost vdora ali zlorabe enaka verjetnosti ugotovitve gesla PIN. Nivo varnosti lahko enostavno dvignemo z daljšim geslom PIN, pri čemer lahko geslo PIN podaljšamo do dolžine 8 znakov in s tem verjetnost ugotovitve gesla še dodatno zmanjšamo na $3 \cdot 10^{-8} + 3 \cdot 10^{-8} = 6 \cdot 10^{-8}$ in dvignemo nivo varnosti iz $\approx 3 \cdot 10^{-4}$ na $6 \cdot 10^{-8}$.

5 Zaključek

Sporočila SMS, ki jih vsakodnevno pošiljamo, niso varna. Posledično niso primerna za uporabo v poslovnih aplikacijah dokler ne zagotovimo primernega nivoja varnosti. Opisana sta dva pristopa s tega področja: protokol SMSsec, ki je programska aplikacija zasnovana na Javansekm paketu WMA (angl. Wireless Messaging API) in kartica PK-SIM, zasnovana na fizičnem nivoju, uporablja dodelano kartico SIM, na kateri se izvajajo operacije. Menim, da je uporaba PK-SIM primernejša, saj je varnejša in primernejša za razširitev na množico mobilnih naprav, za kar bi bila potrebna le menjava kartic SIM. Dokler ne zagotovimo varnosti sporočil SMS, jih ni varno uporabljati za pošiljanje občutljivih in osebnih podatkov. Predlog o izboljšanju omenjenih dveh pristopov je podan v članku N. Saxena, N. S. Chaudhari [3] , ki opisuje nadgradnjo obstoječe kartice SIM, primerjavo šifrirnih algoritmov in uporabo šifriranja na eliptičnih krivuljah.

Literatura

[1] J. L. Lo, J. Bishop, J.H.P. Eloff, SMSSec: An end-to-end protocol for secure SMS, Computers & security 27, str. 154–167, 2008.

[2] H. Rongyu, Z. Guolei, C. Chaowen, X. Hui, Q. Xi, Q. Zheng, A PK-SIM card based end-to-end security framework for SMS, Computer Standards & Interfaces 31, str. 629–641, 2009.

[3] N. Saxena, N. S. Chaudhari, SecureSMS: A secure SMS protocol for VAS and other applications, The Journal of Systems and Software 90, str. 138–150, 2014.

[4] C.S. Park, On certificate-based security protocols for wireless mobile communication systems, IEEE Network 11 (5), str. 50–55, 1997.

[5] C.F. Grecas, S.I. Maniatis, I.S. Venieris, Introduction of the asymmetric cryptographyin GSM, GPRS, UMTS and its public key infrastructure integration, Mobile Networks and Applications 8, str. 145–150, 2003.

[6] A. Wangensteen, L. Lunde, I. Jørstad, T. van Do, A generic authentication system based on SIM, International Conference on Internet Surveillance and Protection, Cap Esterel, French Riviera, avgust 2006.

[7] A. Wangensteen, L. Lunde, I. Jørstad, T. van Do, Secured enterprise access with strong SIM authentication, The Enterprise Computing Conference (EDOC), Hong Kong, oktober 2006.

[8] M. Hassinen, SafeSMS — end-to-end encryption for SMS messages, Proceedings of the 8th International Conference on Telecommunications, str. 359–365, junij 2005.

[9] Statistični podatki Dosegljivo: https://onereach.com/blog/45-texting-statistics-that-prove-businesses-need-to-start-taking-sms-seriously/ [Dostopano 20. 01. 2017].