DOI: 10.15199/ELE-2014-178

Protokół ZRTP – telekomunikacja wspomagana kryptografią

dr inż. WOJCIECH NOWAKOWSKI, prof. ndzw.

Instytut Maszyn Matematycznych

dr hab. inż. TOMASZ ADAMSKI, prof. ndzw.

Politechnika Warszawska, Instytut Systemów Elektronicznych, Wydział Elektroniki i Technik Informacyjnych

W 2011 roku Phil Zimmermann, ważna postać współczesnej kryptografii, członek *Internet Hall of Fame,* twórca powszechnie stosowanego w internecie protokołu **PGP** (ang. *Pretty Good Privacy,* całkiem niezła prywatność) zapewniającego poufność poczty elektronicznej oraz Mike Janke, były specjalista od zabezpieczeń US Navy SEAL a także Jon Callas, twórca oprogramowania szyfrującego zawartość dysków twardych, nawiązali współpracę w celu stworzenia pierwszej na świecie łatwej w użyciu i dostępnej dla wszystkich, prywatnej bezpiecznej cyfrowej łączności mobilnej, zarówno głosowej, tekstowej, video, a także transmisji plików. Tak powstała firma *Silent Circle* z główną siedzibą w Genewie i oddziałami w ośmiu krajach. *Silent Circle* jest obecnie uznaną firmą zapewniającą bezpieczną komunikację cyfrową osobom prywatnym, firmom i rządom w ponad 130 krajach, oferując tani i jednocześnie bardzo zaawansowany system oprogramowania, urządzeń i usług. Bazą usług *Silent Circle* jest protokół ZRTP.

VoIP nie jest bezpieczny [1, 2]

Internet nie jest bezpiecznym środowiskiem łączności telefonicznej. Wraz z rozpowszechnianiem się technologii VoIP (ang. *Voice over IP*) zanika obszar telefonii tradycyjnej (PSTN, ang. *Public Switched Telephone Network*), znacznie trudniejszej do podsłuchu przez osoby niepowołane. Połączenia VoIP będą coraz częściej infiltrowane przez np. przestępczość zorganizowaną. W sieciach biurowych może być np. łatwo wprowadzane oprogramowanie typu *spyware*, które jest w stanie przechwytywać korporacyjne połączenia VoIP i ujawniać tym samym na drugim końcu świata informacje poufne i je sprzedawać. Z kolei rozmowy w sieciach GSM szyfrowane są 64-bitowym kodem A5/1, stworzonym w roku 1987. Obecnie złamanie takiego szyfru za pomocą standardowego komputera z dobrą kartą graficzną zajmuje kilka minut, a za pomocą profesjonalnego sprzętu – kilka sekund. Samo fizyczne przechwycenie danych jest bardzo proste – fale radiowe rozchodzą się we wszystkich kierunkach i do przechwycenia sygnału wystarczy np. programowo przestrajany odbiornik radiowy SDR (ang. *Software Defined Radio*). Ponadto rozmowy GSM są szyfrowane na łączu telefon-operator, a potem ponownie szyfrowane w celu wysłania do rozmówcy. Nie więc ma żadnego problemu w tym, żeby operator podsłuchiwał lub rejestrował wszystkie rozmowy. Istnieją, oczywiście, ograniczenia prawne, ale czy można mieć pewność, że powstrzymają one technika zatrudnionego u operatora przed podsłuchiwaniem, gdy będzie mu to potrzebne?

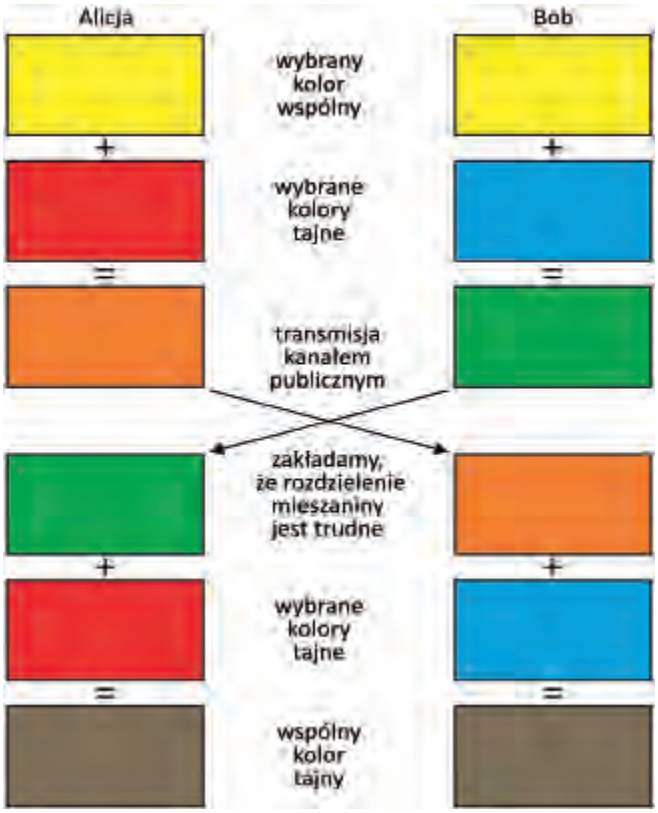
Komunikator *Skype* z kolei, reklamuje się jako bezpieczny, ponieważ realizuje połączenia bezpośrednio między użytkownikami i zabezpiecza je przed podsłuchaniem silnym, 256-bitowym szyfrem. Niestety, ani pierwotni autorzy, ani obecny właściciel – Microsoft – nie zdecydowali się na poddanie publicznym badaniom bezpieczeństwa protokołu komunikacyjnego. Co więcej, Microsoft w miesiąc po zakupie *Skype*'a otrzymał patent na podsłuchiwanie komunikacji w Internecie, a zapytany czy ma możliwość podsłuchiwania rozmów, nie potwierdził ani nie zaprzeczył. Innym popularnym programem do rozmów jest *Google Talk* oraz *Hangouts*. Używają one otwartych protokołów XMPP czy *Jingle*) i są przebadane przez specjalistów. Oba jednak zazwyczaj szyfrują jedynie połączenie klient -serwer, pozwalając pracownikom Google na podsłuchiwanie rozmów.

Warto wreszcie wspomnieć o atakach MITM (ang. *man in the middle* – człowiek w środku). To ktoś, kto ma dostęp do łącza internetowego pomiędzy rozmawiającymi, jak np. Google. Pośrednik ten mógłby przechwytywać pakiety, modyfikować je oraz wysyłać dalej. Zabiegom takim mają zapobiegać certyfikaty SSL, ale jeśli atakujący dysponuje wystarczająco dużymi środkami, może taki certyfikat sfałszowaćcy, jak np. rząd Tunezji, który w czasie arabskiej wiosny, podrobił certyfikaty Facebooka, Gmail oraz Yahoo! i do każdej strony doklejał kod zapisujący wciśnięte klawisze (*keylogger*).

Jaki więc powinien być protokół skuteczny? Powinien mieć przynajmniej dwie cechy: zapewnić takie szyfrowanie danych, aby tylko rozmówcy mogli rozszyfrować strumień danych oraz wykluczyć ataki MITM. Takie właśnie były założenia protokołu ZRTP (zobacz Słowniczek). Bazą tego protokołu jest algorytm Diffiego-Hellmana [3, 4].

Algorytm Diffie'ego-Hellman'a [5]

Algorytm Diffiego-Hellmana (DH)został opracowany przez Witfielda Diffi'ego oraz Martina Hellmana w 1976 roku. Jest on w istocie procedurą uzgadniania wspólnego klucza szyfrująco-deszyfrującego transmisję. Jego siła oparta jest na trudności obliczania logarytmów dyskretnych w ciałach skoń-



Rys. 1. Poglądowy schemat protokołu DH
(***Źródło: A.J. Han Vinck, University of Duisburg-Essen***)
Fig. 1. DH protocol colour scheme
(***Source: A.J. Han Vinck, University of Duisburg-Essen***)

czonych. Algorytm DH pozwala bezpiecznie uzgodnić klucz nawet jeżeli istnieje osoba, która podsłuchuje proces uzgadniania klucza, nie chroni jednak przed atakami typu MITM. Algorytm ten umożliwia wygenerowanie jednego klucza dla obu stron transakcji, bez przesyłania żadnych poufnych informacji. Tak wygenerowany klucz jest później wykorzystywany przez kryptograficzny algorytm symetryczny. Algorytm DH nie jest odporny na atak MITM czyli ingerencję w komunikację między odbiorcą, a nadawcą poprzez podmianę kluczy publicznych. Na *rys. 1* przedstawiono „kolorowo zasadę" działania algorytmu DH.

Poniżej bardziej ogólny opis protokołu:

1. Alicja i Bob wyznaczają dwie liczby: p będącą liczbą pierwsza oraz g (zwany generatorem) mniejsze od p (z następującymi właściwościami: dla każdego $n$pomiędzy 1 i $p-1$ włącznie, istnieje potęga takiego k od g że $n = gk \bmod p$).
2. Alicja generuje prywatną wartość a – Bob generuje prywatną wartość b.
3. Alicja wysyła Bobowi $g^a \bmod p$. Bob wysyła Alicji $g^b \bmod p$. (są to wartości publiczne)
4. Alicja oblicza na podstawie swojej wartości prywatnej $k = (g^b)^a \bmod p$.
5. Bob oblicza na postawie swojej wartości prywatnej $k = (g^a)^b \bmod p$.

Oto przykład najprostszej, protokołu w oryginalnej wersji BH, gdzie publicznie znane wartości oznaczono kolorem niebieskim, a tajne **c**zerwonym:

Alicja				**Bob**		
Tajne	Publiczne	Obliczane	Wysyłane	Obliczane	Publiczne	Tajne
a	p, g		p,g →			b
a	p, g, A	$g^a \bmod p = A$	A →		p, g	b
a	p, g, A		← B	$g^b \bmod p = B$	p, g, A, B	b
a, **s**	p, g, A, B	$B^a \bmod p = s$		$A^b \bmod p = s$	p, g, A, B	b, **s**

Alicja i Bob uzgadniają liczbę pierwszą $p = 23$ i podstawę $g = 5$.

1. Alicja wybiera tajną liczbę całkowitą $a = 6$, i wysyła Bobowi $A = g^a \bmod p$
- $A = 5^6 \bmod 23$
- $A = 15{,}625 \bmod 23$
- $A = 8$

2. Bob wybiera tajną liczbę całkowitą $b = 15$, i wysyła Alicji $B = g^b \bmod p$
- $B = 5^{15} \bmod 23$
- $B = 30{,}517{,}578{,}125 \bmod 23$
- $B = 19$

3. Alicja oblicza $s = B^a \bmod p$
- $s = 19^6 \bmod 23$
- $s = 47{,}045{,}881 \bmod 23$
- $s = 2$

4. Bob oblicza $s = A^b \bmod p$
- $s = 8^{15} \bmod 23$
- $s = 35{,}184{,}372{,}088{,}832 \bmod 23$
- $s = 2$

5. Alicja i Bob współdzielą tajną liczbę: $s = 2$. Jest tak, ponieważ $6*15$ jest tym samym, co $15*6$. Więc jeśli ktoś znałby jednoczenie obie tajne wartości, mógłby także obliczyć s:
- $s = 5^{6*15} \bmod 23$
- $s = 5^{15*6} \bmod 23$
- $s = 5^{90} \bmod 23$
- $s = 807{,}793{,}566{,}946{,}316{,}088{,}741{,}610{,}050{,}849{,}573{,}099{,}185{,}363{,}389{,}551{,}639{,}556{,}884{,}765{,}625 \bmod 23$
- $s = 2$

Zarówno Alicja jak i Bob posiadają tę samą wartość tajną, ponieważ $(g^a)^b$ oraz $(g^b)^a$ są przystające modulo p. Zauważmy że jedynie a, b i $g^{ab} = g^{ba} \bmod p$ są trzymane w tajemnicy. Pozostałe wartości – p, g, $g^a \bmod p$, oraz $g^b \bmod p$ – są wysyłane jawnie. Gdy Alicja i Bob obliczą wspólną wartość, mogą użyć jej jako klucza, znanego tylko im, w publicznym kanae komunikacji. Oczywiście, dla zapewnienia bezpieczeństwa wartości a, b i p powinny być o znacznie większe, ponieważ łatwo jest przeprowadzić próbę dla niewielu kombinacji. Gdy p jest liczbą pierwszą długości ok. 300 cyfr, a a oraz b mają po co najmniej 100 cyfr każda, wtedy nawet najszybszy znany obecnie algorytm na najszybszych komputerach nie znajdzie a mając jedynie g, p, $g^b \bmod p$ i $g^a \bmod p$ w rozsądnym czasie (problem logarytmu dyskretnego). Zauważmy, że g nie musi być duże (w praktyce wybiera się 2 lub 5).

Oboje posiadają teraz element g^{ab}, który może posłużyć jako tajny klucz. W celu odszyfrowania wiadomości m z szyfrogramu mg^{ab}, Bob (lub Alicja) muszę najpierw obliczyć $(g^{ab})^{-1}$: Bob zna $|G|$, b i g^a. Z wartości konstrukcji grupy G, dla każdego x w G, $x^{|G|} = 1$.
Bob oblicza $(g^a)^{|G|-b} = g^{a(|G|-b)} = g^{a|G|-ab} = g^{a|G|}g^{-ab} = (g^{|G|})^a g^{-ab} = 1^a g^{-ab} = g^{-ab} = (g^{ab})^{-1}$.
Kiedy Alicja wysyła Bobowi szyfrogram mg^{ab}, Bob używa $(g^{ab})^{-1}$ i odzyskuje wiadomość $mg^{ab}(g^{ab})^{-1} = m(1) = m$.

Protokół ZRTP (RFC 6189) [6, 5, 7, 8]

Protokół ZRTP (ang. Zimmermann & Real-time Transport Protocol) posiada kilka cech kryptograficznych, których nie ma w wielu innych metodach szyfrowania VoIP. Chociaż używa algorytmu klucza publicznego, nie korzysta z Infrastruktury Klucza Publicznego (PKI). Protokół ZRTP ma opcjonalną możliwość korzystania z infrastruktury PKI, ale wprowadzenie tej architektury do protokołów VoIP jest niekorzystne, co wykazał upadek np. protokołu PEM (zob. Słowniczek).

Protokół ZRTP nie używa stałych kluczy publicznych. Wykorzystuje opisany wyżej algorytm Diffiego-Hellmana z wykorzystaniem także funkcji skrótu *hash*. Ponadto umożliwia wykrycie ataku MITM przez wyświetlanie krótkiego hasła cyfrowego, które użytkownicy mogą porównać ustnie. Posiada również dodatkowe zabezpieczenie przed atakiem MITM, oparte na wprowadzeniu ciągłości transmisji poprzez wykorzystanie cyfrowego fragmentu poprzednie do następnej rozmowy, co daje podstawową ciągłość, podobnie jak czynia to urzędy certyfikacji, Wszystko to nie wymaga PKI, certyfikacji kluczy itp. Protokół ZRTP nie potrzebuje odwoływania się do żadnych serwerów. Uzgadnianie i zarządzanie kluczami tylko *peer-to-peer* za pomocą strumienia pakietów RTP. Automatyczne wykrywa, czy inny klient VoIP obsługuje ZRTP.

Warto zauważyć, że główna różnica między protokołami ZRTP a SRTP (RFC 3711, zob. Słowniczek) polega na tym, że SRTP szyfruje oraz uwierzytelnia wiadomości zapewniając ich integralność i ochronę przed powtarzaniem danych wykorzystując zwykle szyfr symetryczny AES. Ale protokół SRTP nie może być użyty dopóty, dopóki obie strony połączenia nie wynegocjują klucza sesyjnego. System łączności *Silent Circle* opiera się na protokole ZRTP, który wykorzystuje protokół SRTP, ale dopiero po wykonaniu procedury uzgodnieniu klucza. Połączenie ZRTP składa się więc trzech faz. W fazie pierwszej, zostaje zestawione nieszyfrowane połączenie pomiędzy rozmówcami i programy używane do rozmowy sprawdzają, czy druga strona obsługuje szyfrowanie połączeń. Jeśli tak, rozpoczynane jest szyfrowane połączenie – po zakończeniu fazy drugiej, obaj rozmówcy słyszą się nawzajem, są w połączeniu szyfrowanym, ale nie mogą być pewni, czy nie są ofiarami ataku MITM. W fazie trzeciej, na ekranach obu z nich pojawia się ten sam czteroznakowy kod. Jeśli rozmawiając potwierdzą, że obaj widzą ten sam kod – mogą być pewni, że nikt ich nie podsłuchuje. Jak już wspomniano, na potrzeby każdej rozmowy generowany jest inny, losowy klucz. Zaraz po zakończeniu rozmowy jest kasowany z pamięci. Zauważmy więc, że jeśli nawet ktoś nagra całą rozmowę w postaci zaszyfrowanej i np. ukradnie laptopa z którego ta rozmowa była przeprowadzona, nadal nie będzie w stanie jej odzyskać.

Protokół ZRTP jest otwarty i zaraz po jego publikacji pojawiło się kilka programów, które go wykorzystują. Obecnie obsługiwany jest na każdej liczącej się platformie, zarówno komórkowej, jak i desktopowej. Protokół ZRTP definiuje wyłącznie sposób szyfrowania i z tego powodu może być stosowany razem z praktycznie każdym istniejącym protokołem komunikacji.

Początkowo *start-upowa* firma Silent Circle LLC, w ciagu trzech ostatnich lat rozrosła się do formatu światowego i dostarcza swoje produkty i usługi w 130 krajach. Oferta firmy opiera się na wykorzystaniu własnych protokołów ZRTP (do utajniania on-line transmisji audio i video) oraz SCIMP (do zakodowania transmisji tekstowych):

Silent Phone. Bezpieczne rozwiązanie komunikacji mobilnej dla iOS i Android, która obejmuje usługi głosowe, tekstowe, wideo, przesyłanie plików i wiele więcej.

Silent Text. Automatyczne szyfrowanie wiadomości tekstowych. Zawiera funkcje nagrywania, który niszczy wiadomości po przeczytaniu.

Reinventing Privacy. Dedykowane zastosowania platformy bezpiecznych prywatnych usług łączności peer-to-peer we własnej zastrzeżonej sieci.

Silent Phone For Desktop. Silent Phone w wersji *desktop*.

Silent Circle Management Consol. Konsola internetowa do zarządzania w swojej własnej sieci usługami *Silent Phone* i *Silent Text.*

Rys. 2. Blackphone firmy SGP *(Źródło: www.blackphone.ch)*
Fig. 2. SGP Blackphone *(Source: www.blackphone.ch)*

Ostatnią (czerwiec 2014) i koronną propozycją firmy jest *Blackphone* czyli smartfon opracowany przez specjalnie powołana firmę SGP Technologies (*joint venture* GeeksPhone i Silent Circle), który zapewnia szyfrowanie rozmów telefonicznych, e-maili, tekstów i przeglądania Internetu. *Blackphone* zapewnia dostęp do Internetu za pośrednictwem sieci VPN. Telefon ma nowy system operacyjny *PrivatOS,* który jest rozszerzonej wersją Android 4.4.2 o pakiet narzędzi kryptograficznych. B*lackphone* posiada 4,7-calowy ekran, czterordzeniowy procesor 2 GHz, 16 GB pamięci, 8-megapikselowy aparat i LTE.

Zacytujmy z [10]: *Kiedy 20 lat temu Zimmermann udostępnił kod PGP, społeczność internetowa mogła skorzystać z owoców jego pracy za darmo. Tym razem postanowił zarobić na swoim nowym pomyśle. Prawdopodobnie korzystanie z Silent Circle będzie oznaczało konieczność wykupienia miesięcznego abonamentu w wysokości ok. 20 dolarów. Czy internauci gotowi są płacić za większą prywatność? Na razie badania wskazują, że raczej nie – ale może Zimmermann po raz kolejny pokaże, że potrafi zmieniać świat.*

Słowniczek

- **IETF** (ang. Internet Engineering Task Force) to nieformalne, międzynarodowe stowarzyszenie osób zainteresowanych ustanawianiem standardów technicznych i organizacyjnych w Internecie. Jakkolwiek IETF nie posiada żadnej formalnej władzy to jednak prace, które prowadzi mają decydujący wpływ na kształt przyszłości Internetu. IETF generuje specjalny rodzaj dokumentów zwanych Request For Comments (RFC), w których zawarte są definicje dużej części standardów i wielu protokołów internetowych.
- **IM** (ang. Instant Messenger) – komunikator internetowy, czyli program komputerowy pozwalający na przesyłanie natychmiastowych komunikatów (ang. Instant Messaging) pomiędzy dwoma lub większą liczbą komputerów, poprzez sieć komputerową, zazwyczaj Internet). Od poczty elektronicznej różni się tym, że oprócz samej wiadomości, przesyłane są także informacje o obecności użytkowników, co zwiększa znacznie szansę na prowadzenie bezpośredniej konwersacji. Komunikatory IM bardzo często łączą użytkowników przez serwery, do których przyłączają się, i od których działania są uzależnione. Niekiedy tak skrajnie, że użytkownik skazany jest na reklamy emitowane przez producenta aplikacji.
- **PEM** (ang. Privacy Enhanced Mail), opublikowana w 1993 roku w IETF propozycja podwyższenia prywatności komunikacji mailowej e-mail przy użyciu kryptografii klucza publicznego. Chociaż PEM stał proponowanym standardem IETF, nigdy nie został powszechnie zastosowany. Jedną z przyczyn było to, że protokół PEM wymaga infrastruktury klucza publicznego (PKI). Ponieważ ta hierarchiczna struktura została odrzucona, jako infrastrukturę PKI dla metody szyfrowania PGP (ang. *Pretty Good Privacy*) Phil Zimmermann zaproponował zdecentralizowaną metodę *Web of Trust,* uwierzytelniania osób, a zaufanie do poszczególnych certyfikatów jest sumą podpisów złożonych przez innych uczestników sieci. Wysiłki, aby wdrożyć PEM ostatecznie porzucone w odpowiedzi na potrzeby rozszerzenia do obsługi protokołu MIME.
- **PKI** (ang. *Public Key Infrastructure*). Infrastruktura klucza publicznego – zbiór osób, polityk, procedur i systemów komputerowych niezbędnych do świadczenia usług uwierzytelniania, szyfrowania, integralności i niezaprzeczalności za pośrednictwem kryptografii klucza publicznego i prywatnego i certyfikatów elektronicznych [9]. W szczególności jest to szeroko pojęty kryptosystem, w skład którego wchodzą urzędy certyfikacyjne (CA), urzędy rejestracyjne (RA), subskrybenci certyfikatów klucza publicznego (użytkownicy), oprogramowanie oraz sprzęt. Infrastruktura klucza publicznego tworzy hierarchiczną strukturę zaufania, której podstawowym dokumentem jest certyfikat klucza publicznego. Do podstawowych funkcji PKI należą: weryfikacja tożsamości subskrybentów, wymiana kluczy kryptograficznych, wystawianie certyfikatów, weryfikacja certyfikatów, podpisywanie przekazu, szyfrowanie przekazu, potwierdzanie tożsamości i znakowanie czasem
- **QoS** (ang. *Quality of Service*) czyli jakość usługi to, całość charakterystyk usługi telekomunikacyjnej stanowiących podstawę do wypełnienia wyrażonych i zaspokajanych potrzeb użytkownika tej usługi. Aby zapewnić QoS, stosowane są następujące mechanizmy: kształtowanie i ograniczane przepustowości, zapewnienie sprawiedliwego dostępu do zasobów, nadawanie odpowiednich priorytetów poszczególnym pakietom wędrującym przez sieć, zarządzanie opóźnieniami w przesyłaniu danych, zarządzanie buforowaniem nadmiarowych pakietów, określenie charakterystyki gubienia pakietów, unikanie przeciążeń.
- **RTP** (ang. *Real-time Transport Protocol*) to protokół transmisji w czasie rzeczywistym. Pakiet protokołu RTP zawiera informację o typie przesyłanych danych, numer seryjny oraz znacznik czasu. RTP nie gwarantuje jakości usługi (QoS). Protokół RTP najczęściej używa UDP jako protokołu warstwy transportowej. Żeby zagwarantować QoS, RTP jest używany razem z innymi protokołami jak np. RTSP, SIP, które służą do ustalenia połączenia, zanim dane będą mogły być przesłane za pomocą RTP. RTP jest używane w telefonii internetowej (VoIP: Voice over IP) oraz w telekonferencjach.
- **RTSP** (ang. *Real Time Streaming Protocol*) to protokół poziomu aplikacji, mający za zadanie sterowanie dostarczaniem danych czasu rzeczywistego. Mimo że jest on wręcz powszechnie stosowany w aplikacjach związanych z przesyłaniem danych multimedialnych, nie jest on jeszcze ustanowionym oficjalnie standardem, lecz jedynie jego propozycją (ang. draft) ulegającą ciągłym zmianom i korektom. Protokół RTSP dostarcza użytkownikowi jakby elastycznego szkieletu, bazy, która może być rozwijana i dopasowywana do potrzeb użytkownika, aby umożliwić sterowanie transmisją na żądanie danych czasu rzeczywistego takich jak audio i wideo. Źródła danych mogą zawierać dane dwojakiego rodzaju: materiały odtwarzane „na żywo" oraz gromadzone w bazie danych do późniejszego odtworzenia. Protokół w założeniu jego twórców (m.in. RealNetworks) ma służyć kontroli jednocześnie wielu sesji transmisji danych, dostarczając środki do wyboru kanału

transportowego jak np. UDP, rozgałęziany UDP i TCP oraz środki do wyboru odpowiednich mechanizmów działania opartych na protokole RTP. Protokół RTSP jest rodzajem jakby sieciowego „pilota" (ang. *network remote control*) dla serwerów multimedialnych. W protokole tym w zasadzie nie występuje pojęcie połączenia. Zamiast tego przyjmuje się, że serwer RTSP utrzymuje sesje oznaczoną odpowiednim identyfikatorem, która łączy grupy strumieni mediów i ich stanów. Sesja protokołu RTSP nie jest związana z pojęciem połączenia na poziomie warstwy transportowej w rozumieniu połączenia TCP. Podczas sesji użytkownik może otwierać i zamykać wiele pewnych (w znaczeniu niezawodnych) połączeń transportowych z serwerem, aby wysyłać żądania protokołu RTSP dla tej sesji. Protokół RTSP może używać protokołu transportowego TCP gwarantującego niezawodne połączenie lub niepewnego bezpołączeniowego protokołu transportowego UDP. Strumienie sterowane przez protokół RTSP mogą używać protokołu RTP do transportu swoich danych. Protokół RTSP jest podobny jest do protokołu HTTP (ang. HyperText Transfer Protocol), ale i wyraźnie od niego się różni.

- **SCIMP** (ang. *Silent Circle Instant Messaging Protocol*) protokoł internetowy firmy Silent Circle, który zapewnia silne szyfrowanie, utajnienie i uwierzytelnianie komunikatów natychmiastowych (IM, ang. *Instant Messages*). SCIMP wykorzystuje szereg algotytmó i protokołów, w tym ZRTP, OTR (*Off The Record*), SSMS (*Secury Short Message Service*), Cryptocat NADM, krzywe eliptyczne Diffiego-Hellmana (ECDH) raz szereg standardów NIST.
- **SIP** (ang. *Session Initiation Protocol*) to protokół inicjowania sesji, zaproponowanym przez IETF jako standard dla zestawiania sesji pomiędzy jednym lub wieloma klientami. Jest obecnie dominującym protokołem sygnalizacyjnym dla Voice over IP i stopniowo zastępuje H.323. SIP ma w zamierzeniu dostarczać zestaw funkcji obsługi połączenia i innych cech obecnych w publicznej sieci telefonicznej (PSTN). Jako taki zawiera funkcje, które umożliwiają znane ze stacjonarnej telefonii operacje: wybieranie numeru, dzwonek w telefonie, sygnał zajętości, chociaż ich implementacja jest odmienna. Istnieje wiele innych protokołów sygnalizacyjnych dla VoIP, jednakże SIP zdefiniowano pośród społeczności internetowej, a nie telekomunikacyjnej. SIP jest standardem zarządzanym przez IETF. Starsze i bardziej złożone protokoły VoIP były zazwyczaj propozycjami zgłaszanymi przez ITU-T. SIP jest podobny do HTTP i dzieli z nim wiele zasad konstrukcyjnych: używa zwykłego tekstu (jest możliwy do czytania bezpośrednio przez człowieka), bardzo prosty mechanizm żądanie-odpowiedź.
- **SRTP** (ang. *Secure Real-time Transport Protocol*) to protokół RTP z szyfrowaniem, uwierzytelnianiem wiadomości i zapewnianiem integralności oraz ochrony przed powtarzaniem danych zarówno w trasmisji unicast (dokładnie jeden punkt wysyła pakiety do dokładnie jednego punktu) jak i multicast (gdy liczba odbiorców jest dowolna). Został on opracowany przez ekspertów kryptograficznych firm Cisco i Ericsson. Po raz pierwszy został opublikowany przez IETF w 2004 roku (RFC 3711). Do szyfrowania i deszyfrowania strumienia danych domyślnym szyfrem SRTP jest AES. Ten szyfr blokowy może w innym trybie pracować jako szyfr strumieniowy.
- **UDP** (ang. *User Datagram Protocol*) to protokół pakietów użytkownika, jeden z protokołów internetowych. UDP stosowany jest w warstwie transportowej modelu OSI, czyli standardu zdefiniowanego przez ISO oraz ITU-T opisującego strukturę komunikacji sieciowej.

ZRTP (ang. *Zimmermann Real-time Transport Protocol*) jest protokołem kryptograficznym służącym do uzgodnienia kluczy niezbędnych do szyfrowania transmisji między dwoma punktami końcowymi w telefonii VoIP (ang. *Voice over Internet Protocol*) na podstawie protokołu RTP. Protokół ZRTP wykorzystuje protokół uzgadniania i wymiany kluczy, opracowany przez Witfielda Diffiego oraz Martina Hellmana oraz protokół SRTP do bezpiecznej transmisji danych w czasie rzeczywistym. Protokół ZRTP został opracowany przez Phila Zimmermanna, z udziałem zespołu (Bryce Wilcox-O'Hearn, Colin Plumb, Jon Callas i Alan Johnston) i opublikowany w kwietniu 2011 (RFC 6189). Protokół ZRTP został wdrożony i jest stosowany na następujących platformach: Windows, Linux, MacOSX, iPhone, Symbian, BlackBerryOS, Android, w następujących językach: C, C++, Java oraz w następujących typach transmisji danych: WiFi, UMTS, EDGE, GPRS, satelitarny IP, GSM CSD, ISDN.

Literatura

[1] http://silentcircle.com

[2] http://websecurity.pl/zrtp-bezpieczna-komunikacja-glosowa/

[3] http://pl.wikipedia.org/wiki/Protok%C3%B3%C5%82_Diffiego-Hellmana

[4] http://www.algorytm.org/inne/algorytm-diffie-hellmana.html

[5] http://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange

[6] http://zfone.com/zrtp_ietf.html

[7] http://websecurity.pl/silent-circle-wprowadza-usluge-szyfrowanych-rozmow-dla-androida/

[8] http://en.wikipedia.org/wiki/ZRTP

[9] http://www.ecb.europa.eu/ecb/legal/pdf/l_07420130316pl00300035.pdf

[10] http://zaufanatrzeciastrona.pl/post/tworca-pgp-da-nam-prywatnosc-rozmow-telefonicznych