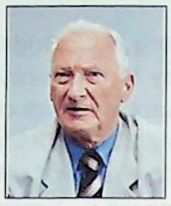
Brunon Hołyst, Jacek Pomykała



Bezpieczeństwo podpisu progowego w grupach dynamicznych

Wiedza na temat tradycyjnego podpisu cyfrowego jest w społeczeństwie informacyjnym powszechna. W gospodarce elektronicznej stosowane są różne schematy podpisu, np. standardy ANSI, takie jak DSA[1], RSA[2] czy EC-DSA[3]. Wiele rozwiniętych krajów świata wprowadziło odpowiednie regulacje prawne[4], zaś w niektórych państwach europejskich, np. skandynawskich, sprawnie funkcjonuje już infrastruktura podpisu elektronicznego.

Tradycyjny podpis cyfrowy poddano głębokiej analizie i dokonano jego klasyfikacji w kategoriach bezpieczeństwa. Aspekty kryminalistyczne, prawne i informatyczne były zaś przedmiotem badań omówionych we wcześniejszym artykule[5]. Niniejszy tekst jest jego kontynuacją i ma ukazać nowe rozwiązania, które dotyczą bezpieczeństwa podpisu progowego w grupach dynamicznych – zagadnienie szczególnie aktualne w odniesieniu do współczesnych aplikacji internetowych i komunikacji w sieciach bezprzewodowych lub sieciach sensorów.

Kwestia bezpieczeństwa podpisu progowego wymaga niezbędnych zmian i uściśleń w odniesieniu do pojęcia autentyczności podpisu. W podpisie tradycyjnym oznacza ona, że podpisujący jest identyfikowalny. W podpisie progowym „autentyczność" nie sprowadza się już do identyfikacji podpisujących, zatem w tym nowym modelu podpisu należy doprecyzować pojęcie fałszerstwa.

Z fałszerstwem mamy do czynienia wtedy, gdy poprawny podpis zostaje wygenerowany przez podgrupę członków (na ogół nieidentyfikowalnych) o liczbie mniejszej niż wymagana liczba progowa. Po drugie, dochodzi tu dodatkowy wymiar bezpieczeństwa – członkowie skorumpowani nie są w stanie uniemożliwić uczciwym członkom podgrupy złożenie podpisu. Pozostałe aspekty bezpieczeństwa można zdefiniować w sposób tradycyjny, określając fałszerstwo jako odpowiednią funkcję typu ataku i poziomu sukcesu potencjalnego przeciwnika. Poziom sukcesu ataku prowadzi do wyodrębnienia następujących kategorii: fałszerstwo egzystencjalne, selektywne, uniwersalne i totalne. Ten model bezpieczeństwa tradycyjnego podpisu opisano dokładnie w wielu podręcznikach[6]. O wymiarze grupowym podpisu cyfrowego wspomniano zaś w poprzednim artykule autorów niniejszego tekstu[7]. Poniżej rozwiniemy ten temat, przeprowadzając analizę podpisu progowego dla tzw. grup dynamicznych.

Kryptografia progowa – podstawowe pojęcia i definicje

Głównym wyzwaniem dla kryptografii progowej jest ochrona informacji w ramach systemów z tolerancją błędów (*fault tolerantly distributing*). Największym problemem jest kwestia bezpiecznego dzielenia sekretu (*secure sharing of a secret*), co polega na rozdzieleniu części sekretu (tajnej informacji) między wiele podmiotów (np. serwerów) tak, aby zostały spełnione następujące wymagania:

1) żadna grupa skorumpowanych graczy (członków grupy) w liczbie mniejszej niż progowa nie jest w stanie zrekonstruować na podstawie tych części tajnej informacji nawet w przypadku, gdy skorumpowani gracze współpracują ze sobą,

2) w razie potrzeby rekonstrukcji tajnej informacji może dokonać dowolna, dostatecznie duża grupa (co najmniej w liczbie progowej) uczciwych graczy.

Dzielenie sekretu jest kluczowym elementem schematu podpisu progowego. Załóżmy, że dany jest schemat podpisu cyfrowego (K, S, V), gdzie K jest algorytmem generacji kluczy, S algorytmem podpisywania, zaś V algorytmem weryfikacji podpisu. Odpowiedni schemat podpisu progowego (TK, TS, V) różni się dwoma pierwszymi algorytmami. TK jest algorytmem dystrybucji udziałów klucza, który na wejściu ma dane publiczne członków grupy, natomiast na wyjściu klucz publiczny grupy pk i odpowiednie udziały (prywatne) członków grupy. Wymagane jest, aby rekonstrukcja tych udziałów (zgodnie z punktem 2 powyżej) prowadziła do odpowiedniego klucza prywatnego sk danej grupy, takiego, że para (sk, pk) jest parą odpowiednią dla algorytmu K. Co więcej, rozkład takich par dla algorytmu K jest taki sam jak dla algorytmu TK. Spełnienie takiego wymagania eliminuje możliwość ataków polegających na tzw. sterowaniu kluczem. Algorytm generacji podpisu progowego TS składa się z dwóch etapów: generacji podpisów częściowych i fazy rekonstrukcji podpisu. W pierwszym etapie każdy z graczy na podstawie wiadomości i swojego udziału oblicza odpowiedni podpis częściowy, który w etapie drugim zostaje złożony do

pełnego podpisu progowego grupy. Wymaganiem dla algorytmu *TS* jest to, by dla dowolnej wiadomości złożony w ten sposób podpis był taki sam jak podpis otrzymany z użyciem algorytmu *S*.

Celem kryptografii progowej jest implementacja wydajnych protokołów gwarantujących najwyższy poziom bezpieczeństwa odpowiednich systemów kryptograficznych w możliwie silnych modelach z tolerancją błędów. Realizację takiego celu osiąga się z uwzględnieniem następujących warunków:

a) dopuszczalny poziom korupcji, który nie powoduje szkody w działaniu całego systemu,
b) wymagania wydajności protokołu, tj. zasoby (pamięć), koszty komunikacji i koszty obliczeń,
c) przyjęty model komunikacji, tj. synchroniczność komunikacji, istnienie poufnych kanałów komunikacji między członkami grupy, istnienie publicznego, uwierzytelnionego kanału informacyjnego (*broadcast channel*),
d) typ przeciwnika określający, w jaki sposób może on korumpować graczy i jakie są możliwości obrony gracza przed przeciwnikiem (np. czy potrafi skutecznie zlikwidować posiadane zasoby, by nie dopuścić do ich infiltracji przez atakującego),
e) identyfikacja zagrożeń, jakie mogą pojawić się w wyniku implementacji takich protokołów zarówno na platformie programowej, jak i sprzętowej.

W dalszej części pracy zajmiemy się dokładniej schematami deszyfrowania oraz podpisu progowego w grupach niestatycznych.

Grupy dynamiczne

Pojęcie „grupa dynamiczna" odnosić będziemy nie tylko do zmienności liczby jej członków (dodawanie nowych, usuwanie skompromitowanych), lecz także zmienności ich tożsamości lub pełnionych przez nich funkcji. Przykładem dynamiki grupowej jest zmiana przynależności członka z jednej podgrupy do drugiej (w ramach jednej rodziny grup). Dynamika może wymuszać zmiany sekretu grupy lub nie. Ponieważ generowanie nowego sekretu grupy jest zazwyczaj związane z ponowną inicjalizacją systemu, takie rozwiązania są w praktyce mało wydajne. Z tego powodu nie będziemy poświęcać im uwagi. Załóżmy więc, że dana jest a priori niewielka grupa *G*, której członkowie mają udziały w sekrecie grupy *s*.

Niech zbiorem udziałów dla tajemnicy *s* będzie zbiór $\{s_1, s_2,..., s_m\}$. Jeśli do rekonstrukcji sekretu *s* niezbędne są wszystkie udziały s_i, będziemy mówić o rozkładzie (*splitting*) sekretu *s* na udziały s_i, jeśli zaś wystarczy ich progowa liczba *t* ($t < m$), to powiemy o progowym dzieleniu (*threshold secret sharing*) tajemnicy. Będziemy dalej zakładać, że grupa może zrekonstruować tajemnicę *s*, tylko z użyciem co najmniej *t* wartości udziałów. Typowym przykładem rozkładu sekretu jest losowy wybór $m-1$ udziałów i zdefiniowanie $s_m = s - (s_1 + s_2 +... s_{m-1})$. Jest to tzw. rozkład addytywny (dzielenie addytywne) w odróżnieniu od dzielenia wielomianowego (interpolacyjnego).

Wzorcowym przykładem dzielenia progowego jest schemat Shamira[8] oparty na interpolacji Lagrange'a. Interpolowany wielomian ma wtedy stopień $t-1$, gdzie *t* jest wartością progu. Jakkolwiek rozkład sekretu można zrealizować jako skrajne dzielenie tajemnicy ($t = m$), nie jest to rozsądne. Znaczenie schematów progowych pojawia się dla $t < m$, jak zobaczymy w dalszej części artykułu.

Według standardowego scenariusza realizacji dynamiki do grup dołącza się nowych członków. W trakcie tego procesu może się okazać, że w grupie wyodrębniają się specyficzne podgrupy (np. ze względu na specyfikację zadań wykonywanych przez jej członków). Jeśli od tak wyspecjalizowanych podgrup będziemy oczekiwać autoryzacji określonych wiadomości, to będziemy mieć do czynienia z podpisami cyfrowymi członków w imieniu tych podgrup. Jeśli dodatkowo zbiory autoryzujące zawierają minimalne zbiory (progowe), mówi się o podpisach progowych dla danej rodziny grup.

Dołączanie nowych członków może odbywać się „lokalnie" (tzn. przez pojedynczych graczy już istniejących w grupie) lub globalnie, tj. za wiedzą i zgodą quorum grupy. Jest przy tym jasne, że proces dołączania nowych członków można realizować, wykorzystując jeden z dwóch typów podziału tajemnicy omówionych powyżej. Przypadek pierwszy badany w artykule R. Di Pietra, L.V. Manciniego i G. Zanina[9] można rozszerzyć także do badania systemów pełnomocnictw cyfrowych, którym poświęcona była praca J. Pomykały i S. Barabasza[10]. W modelu grupowym ma to oczywisty związek z grupami hierarchicznymi[11]. Przypadek globalny rozważano zaś w artykule J. Pomykały i T. Warchoła[12].

Dynamika grup może być realizowana na różne sposoby w zależności od wymagań stawianych systemowi. Przykładowo, dynamika reprezentacji grup w obrębie większych rodzin została omówiona w powyższej pracy J. Pomykały i T. Warchoła dzięki wprowadzeniu odpowiednich certyfikatów cyfrowych dla członków grupy. Ważnym elementem bezpiecznego funkcjonowania grup dynamicznych są przyjęte założenia, np. dotyczące istnienia zaufanych stron w protokołach, takich jak: diler rozdzielający udziały dla członków grupy, administrator przydzielający certyfikaty czy menedżer weryfikujący poprawność udziałów lub certyfikatów. Im słabsze założenia, tym lepszy jest system

– pod warunkiem, że umie się dowieść jego bezpieczeństwa.

Idea podpisu progowego została wprowadzona przez Y. Desmedta[13], a jego bezpieczny schemat dla grupy statycznej zaproponowali R. Genarro, S. Jarecki, H. Krawczyk i T. Rabin[14]. Bezpieczny schemat progowy podpisu dla rodziny grup dynamicznych w minimalnym modelu zaufania został omówiony we wspomnianej pracy J. Pomykały i T. Warchoła[15], zaś kwestię alternatywnego modelu podpisu progowego dla grupy dynamicznej poruszyli w swej pracy R. Di Pietro, L.V. Mancini i G. Zanin[16]. Tego typu rozwiązanie jest szczególnie dogodne dla bezpiecznego gromadzenia i przekazywania informacji w rozległych sieciach sensorowych[17].

Kryptosystemy progowe

Kryptosystemy progowe realizują bezpieczną i wydajną wymianę informacji w strukturach grupowych. Dwie kluczowe aplikacje są związane z podpisem progowym i tzw. deszyfrowaniem progowym w zależności od tego, czy progowość jest z kolei związana z grupowym szyfrowaniem lub deszyfrowaniem. W obu sytuacjach progowość jest szczególnym przypadkiem tzw. ogólnych struktur dostępu (*access structures*), w których do poprawnego zaszyfrowania lub odszyfrowania wiadomości potrzebna jest grupa uprzywilejowana, będąca w tym przypadku dowolną grupą posiadającą progową liczbę udziałów w sekrecie grupy. Najbardziej typowymi schematami progowymi są tzw. (t, n) schematy, gdzie n oznacza liczbę członków grupy, natomiast t liczbę udziałów niezbędnych do odtworzenia sekretu grupy (wartość progu). Głównym powodem tworzenia tego typu systemów kryptograficznych jest zapewnianie niezawodności i wydajności usług w systemach teleinformatycznych, zamiar zwiększenia zaufania podmiotów świadczących usługi, a także rozdzielanie odpowiedzialności w uzgadnianiu odpowiednich decyzji. W praktycznej realizacji protokołu istotnym zadaniem jest ustalenie poziomu progu dobieranego odpowiednio do zakładanego poziomu korupcji. Systemy deszyfrowania progowego typu (t, n) odgrywają ważną rolę w sytuacjach, gdy odszyfrowanie wiadomości nie może być zależne tylko od posiadania jednego klucza deszyfrującego. Stwarzałoby to ryzyko, że w wypadku jego utracenia informacja nigdy nie zostałaby odzyskana. Z drugiej strony, w przypadku gdy szyfrowane informacje mają np. klauzulę „ściśle tajne", może być istotne, aby ich zdeszyfrowanie mogło być przeprowadzone w obecności kilku stron.

Omówimy teraz ideę deszyfrowania progowego z wykorzystaniem systemu RSA. Pomysł pochodzi od J. Pieprzyka i wsp.[18] i ma następujące zalety:

a) system nie wymaga istnienia zaufanej strony,
b) system nie wymaga przekazywania udziałów podpisu bezpiecznymi kanałami,
c) każdy użytkownik jest wyposażony w parę kluczy: prywatny i publiczny, przy czym klucz publiczny jest przechowywany w pewnym publicznym rejestrze (tzw. *white pages*),
d) docelowa grupa odbiorców wiadomości oraz poziom progu odbiorców są wybierane przez nadawcę,
e) kryptogram może być odczytany jedynie przy współpracy progowej liczby odbiorców z ustalonego przez nadawcę zbioru.

Protokół składa się z dwóch algorytmów:

- szyfrującego, który na wejściu ma klucze publiczne użytkowników, wiadomość m i próg t, a na wyjściu kryptogram, który jest podany do wiadomości publicznej,
- deszyfrującego, który mając na wejściu kryptogram i klucze prywatne dowolnej progowej liczby uczestników, oblicza wartość odszyfrowanej wiadomości.

W pierwszym etapie deszyfrowania nadawca oblicza iloczyn wykładników publicznych odpowiedniego zbioru użytkowników, do którego adresowana jest wiadomość oraz iloczyn N ich modułów RSA. Następnie losuje wielomian f stopnia $t–1$ nad ciałem skończonym, którego wyraz wolny s jest sekretem odbierającej grupy, po czym dokonuje lokalizacji:

a) przygotowaną wiadomość M (mniejszą od iloczynu odpowiednich modułów) reprezentuje lokalnie w postaci reszt modulo, a odpowiednie publiczne moduły odbiorców i odpowiednie reszty podnosi lokalnie do potęgi s,
b) sekret grupy dzieli wielomianem f, otrzymując odpowiedni ciąg udziałów (lokalna reprezentacja tajemnicy), który szyfruje następnie za pomocą wykładników publicznych odbiorców.

Tak uzyskane ciągi „skleja" modularnie, otrzymując globalną parę (C_1, C_2). Wyjściem fazy szyfrowania jest czwórka (N, t, C_1, C_2).

W drugim etapie członkowie grupy najpierw lokalizują parę (C_1, C_2), a następnie, używając swoich kluczy prywatnych RSA, obliczają na podstawie C_2 w grupie progowej wartość s, którą podają do wiadomości publicznej. Teraz każdy odbiorca odtwarza lokalnie wiadomość M, korzystając ze znajomości zapadki dla „swojego" systemu RSA. Ostatnią fazą jest „sklejenie" wiadomości m na podstawie jej wartości lokalnych (reszt), które może się odbyć w dowolnej grupie progowej.

Podpis progowy to w istocie szyfrowanie progowe kluczem prywatnym. Składa się z trzech algorytmów:

a) algorytmu generacji kluczy będącego interaktywnym protokołem przeprowadzanym przez człon-

ków grupy, który na wejściu ma dane publiczne grupy, a na wyjściu daje klucz publiczny grupy wraz z odpowiednim udziałem dla każdego użytkownika indywidualnie. Rekonstrukcja tych udziałów prowadzi do klucza prywatnego grupy będącego w odpowiedniości z wygenerowanym kluczem publicznym,

b) algorytmu podpisywania mającego na wejściu wiadomość i klucze prywatne podpisujących, z których najpierw generuje odpowiednie udziały podpisu (faza generacji podpisów częściowych), a następnie łączy je w cały podpis (faza rekonstrukcji podpisu),

c) algorytmu weryfikacji podpisu, który mając na wejściu klucz publiczny grupy i podpis pod zadaną wiadomością, daje na wyjściu odpowiedź: akceptuj lub odrzuć.

Ogólny przegląd kryptografii progowej został omówiony przez jednego z jej czołowych twórców – Y. Desmedta w pracy „Threshold cryptography"[19]. Ze względu na powszechnie funkcjonujące obecnie modele zaufania w komunikacji elektronicznej trudno jest przecenić rolę bezpieczeństwa danych i zastosowania protokołów wykorzystujących obliczenia wielopodmiotowe. W następnym paragrafie zdefiniujemy koncepcję podpisu wieloprogowego, który odpowiada na nowe wyzwania współczesnej komunikacji elektronicznej.

Podpis wieloprogowy

Jak wykazaliśmy, można efektywnie zaprojektować system deszyfrowania, w którym wartość progu jest ustalana razem z przygotowaną do zaszyfrowania wiadomością. W przypadku szyfrowania kluczem prywatnym grupa jest stroną autoryzującą wiadomość. W takiej sytuacji, zmieniając próg dla danej wiadomości, wymusza się też zmianę wielomianu dzielącego sekret grupy. Z punktu widzenia wydajności jest to duże osłabienie systemu, gdyż wymaga częstego zaangażowania generatora liczb pseudolosowych przy podpisywaniu różnych wiadomości (z różnym progiem). Rozwiązanie zaproponowane w pracy B. Nakielskiego, J. Pomykały i J.A. Pomykały[20] jest następujące: skoro nie chcemy zmieniać wielomianu każdorazowo przy podpisywaniu różnych wiadomości, to ustalmy wielomian wyższego stopnia, odpowiadający większej grupie, której podgrupa będzie pełniła rolę podpisujących. Prowadzi to do wyodrębnienia dwóch grup: podpisujących i uzupełniających.

Z punktu widzenia dużej grupy mamy do czynienia ze zwykłym podpisem progowym, natomiast z punktu widzenia jej podgrupy z podpisem o zmiennym progu, czyli wieloprogowym. Taka elastyczność w wyborze progu może wskazywać na potencjalne zastosowania np. dla podpisów wiadomości o różnych „priorytetach". Co więcej, powyższa idea pozwala na realizację zarówno podpisu anonimowego, jak też z pełną identyfikacją podpisujących. Drugi wariant można osiągnąć przez prostą autoryzację podpisu „anonimowego" z użyciem prywatnych kluczy podpisujących. Schemat zaproponowany w powyżej cytowanej pracy składa się z czterech następujących faz:

a) algorytmu inicjalizacji systemu,
b) fazy podpisywania,
c) fazy autoryzacji,
d) fazy weryfikacji podpisu.

Omówiony powyżej podpis wieloprogowy bazuje na systemie RSA i protokole Shamira dzielenia sekretu. W kolejnym paragrafie wprowadzimy nową ideę zastosowania podpisów progowych dla rodziny grup dynamicznych z użyciem systemu certyfikatów cyfrowych.

Podpisy progowe dla grupy dynamicznej

Można wyróżnić dwa kluczowe i całkowicie odmienne podejścia do realizacji takich schematów. Każde z nich inaczej reprezentuje ideę progowości. Jedno odnosi ją do dystrybucji niepowtarzalnych udziałów dla poszczególnych członków grupy, drugie zaś do dystrybucji powtarzalnych udziałów w grupie. W obydwu przypadkach liczba różnych udziałów decyduje o właściwej liczbie progowej.

Zacznijmy od drugiej idei zaprezentowanej np. w pracy R. Di Pietra, L.V. Manciniego i G. Zanina[21]. Podstawowym powodem zastosowania tego pomysłu do sieci mobilnych są następujące uwarunkowania:

a) dynamiczne uczestnictwo w podpisywaniu,
b) dynamiczna struktura grupy,
c) ograniczoność zasobów,
d) rozproszenie obliczeń,
e) brak zaufanej infrastruktury,
f) potrzeba świadczenia usług i realizacji protokołów grupowych,
g) minimalne założenia dotyczące operacji w węzłach sieci (np. wyposażenie każdego węzła w generator liczb pseudolosowych i niewielkie wymagania co do lokalnych zdolności obliczeniowych).

Udziały członków są rozdzielane w grupie z rozkładem równomiernym (można to osiągnąć dzięki zastosowaniu odpowiedniej funkcji mieszającej). Do wygenerowania podpisu potrzeba zbioru wszystkich różnych udziałów rozproszonych w grupie. Jeśli wiele udziałów jest powtarzalnych (równych), to istnieje dużo odpowiednich podgrup progowych. Zapewnia to więc przede wszystkim ciągłość realizacji usług przez odpowiedni system. Z drugiej strony przeciwnik może doprowadzić do zablokowania działania systemu jedynie w razie

przejęcia kontroli nad całą podgrupą członków o jednakowych udziałach. Zatem w przypadku, gdy członków całej grupy jest znacznie więcej niż różnych udziałów, system ma wysoki poziom bezpieczeństwa. Probabilistyczna analiza wydajności przeprowadzona w pracy dowodzi także jego przydatności praktycznej.

Udziały dla nowych członków grupy są generowane przez replikację jednej klasy (członków o jednakowych udziałach) przy współudziale jakiejkolwiek innej klasy koniecznej do aktualizacji danych w oparciu o pomysł addytywnego dzielenia tajemnicy. Mocną stroną tego protokołu jest fakt, że dynamika (dołączanie nowych graczy) odbywa się przez replikację przy zaangażowaniu jedynie dowolnych dwóch członków grupy z różnych klas.

Pierwsza idea w grupie dynamicznej wykorzystuje statyczny schemat wielomianowego dzielenia tajemnicy[22] powołujący się na interpolację Lagrange'a. Wszystkie rozdzielane udziały są różne, zatem progowość jest z całkowitą pewnością (a nie tylko prawdopodobieństwem) zagwarantowana przez liczebność odpowiedniej podgrupy. Mocną stroną schematów opartych na tej idei są:

a) orientacja na progowość liczby podpisujących, a nie liczby różnych udziałów,
b) większa elastyczność w generowaniu sekretu grupy (dzielenie wielomianowe, a nie addytywne),
c) rozdzielenie odpowiedzialności podpisujących,
d) zapewnienie ciągłości podpisywania (eliminacja wąskich gardeł),
e) wzmocnienie zaufania użytkowników,
f) rozproszenie obliczeń i podwyższenie wydajności systemu,
g) możliwość realizacji w grupach dynamicznych.

Ideę dynamiki w rodzinie grup zaproponowaną we wspomnianej już pracy J. Pomykały i T. Warchoła[23] można wyrazić następująco: mamy zadaną rodzinę, której członkowie zaczynają się dzielić na określone kategorie (np. ze względu na specyfikę zadań, jakie będą wykonywać). Aby wyróżnione grupy mogły podpisywać wiadomości, wprowadzamy certyfikaty przynależności do odpowiednich rodzin. Zakłada to istnienie dynamiki w zakresie wymienności ról członków całej rodziny. Aby wzmocnić taki schemat o elastyczność wyboru progu, w omawianej pracy zastosowano pomysł podpisu wieloprogowego. Niewątpliwą przewagą tego schematu w porównaniu do schematu opartego na addytywnym dzieleniu sekretu jest fakt, że – pomijając członków skorumpowanych – każda podgrupa w ilości progowej może poprawnie podpisać zadaną wiadomość. Po drugie, przy wysokim poziomie progu dla liczby koniecznych udziałów system ten – w przeciwieństwie do protokołu opisanego w pracy[24] – jest całkowicie bezpieczny.

Analiza bezpieczeństwa

Jak już pisaliśmy[25], bezpieczeństwo klasycznego podpisu cyfrowego definiuje się w odniesieniu do charakteru przeciwnika (*adversary*). Dokładniej musimy określić więc jego cel i środki, jakimi dysponuje. W obecnej sytuacji przyjmujemy, że przeciwnik jest w stanie przejąć kontrolę (skorumpować) nad – co najwyżej – k-graczami. Oznacza to przede wszystkim, że skorumpowani uczestnicy protokołu mogą się zachowywać tak, jak wymaga tego adwersarz, z przekazaniem mu swoich kluczy prywatnych włącznie. Zakładamy, że w analizowanym modelu obliczeniowym przeciwnik dysponuje ograniczoną (wielomianową) pamięcią i mocą obliczeniową. Celem adwersarza jest działanie, które doprowadzi do sfałszowania podpisu progowego lub nie doprowadzi do wygenerowania podpisu, mimo że w grupie istnieje progowa liczba uczestników uczciwych.

Dowód bezpieczeństwa schematu polega na przedstawieniu redukcji złamania takiego schematu progowego do złamania klasycznego schematu podpisu[26] w zadanym modelu. Udowodnienie, że przeciwnik nie jest w stanie osiągnąć celu (z niepomijalnym prawdopodobieństwem) będzie oznaczać, że podpis jest niepodrabialny (*unforgeable*) i odporny (*robust*). Podrabialność będąca tu synonimem fałszerstwa oznacza, że adwersarz jest w stanie wygenerować podpis, który przejdzie pomyślnie weryfikację, choć nie ma progowej liczby udziałów podpisu od uczestników skorumpowanych.

Jeśli progiem jest t, a liczbą skorumpowanych k, oznacza to, że jeśli podpis jest poprawny, to co najmniej $t - k$ uczciwych (tj. nieskorumpowanych) członków grupy musiało uczestniczyć (zaangażować swoje udziały) w protokole. Jeśli wymagamy, aby liczba uczciwych podpisujących była co najmniej l, to otrzymujemy stąd nierówność $t > k + l - 1$ określającą wysokość progu, który gwarantuje niepodrabialność podpisu. Z drugiej strony odporność schematu implikuje, że wśród n członków grupy będzie co najmniej tylu uczciwych, ilu potrzeba do wygenerowania poprawnego podpisu, tj. co najmniej k. Stąd otrzymujemy drugi warunek bezpieczeństwa (*robustness*): $n - k > t - 1$.

Poziom korupcji k nie może więc przekroczyć połowy liczby $n - l$. Przypadek, kiedy to l może być istotnie większe niż 1 był rozważany w pracy V. Shoupa[27]. Udowodniono w niej, że otrzymany schemat jest bezpieczny w modelu z losową wyrocznią, odwołując się do bezpieczeństwa klasycznego systemu podpisu RSA[28] (por. także rozdz. 8.5)[29]. W artykule J. Pomykały i T. Warchoła[30] omówiono zaś zagadnienie redukcji bezpieczeństwa podpisu progowego w grupie dynamicznej, w odniesieniu do podpisu progowego w grupie

statycznej[31]. Warto dodać, że zastosowanie arytmetyki krzywych eliptycznych implikuje tu redukcję bezpieczeństwa systemu do potencjalnie trudniejszego problemu logarytmu dyskretnego na krzywej eliptycznej niż w grupie modularnej.

Podsumowanie

W niniejszej pracy skoncentrowano się na kwestii bezpieczeństwa progowego podpisu cyfrowego w grupach dynamicznych. Wnioski płynące z przeprowadzonej analizy są następujące: charakteryzując rozważane schematy czterema parametrami: (n, m, t, k), gdzie n oznacza wielkość grupy w danym momencie, m liczbę różnych udziałów w grupie, t progową liczbę udziałów, zaś k maksymalną liczbę skorumpowanych członków, stwierdzamy, że schematy wykorzystujące dzielenie addytywne odpowiadają przypadkowi, gdy $m = t$, a schematy wykorzystujące dzielenie wielomianowe przypadkowi, gdy $m = n$. Analiza bezpieczeństwa przekonuje, że pierwszy typ powinien być preferowany dla małych wartości t, gdyż wtedy mamy dobre ograniczenie na poziom korupcji: $k < n/t$, natomiast drugi dla dużych wartości t, gdyż wtedy mamy ograniczenie: $k < t$. Ostatecznie wybór typu schematu zależy od ściśle przeprowadzonej analizy wymagań, bezpieczeństwa i wydajności funkcjonalnej systemu.

PRZYPISY

[1] DSA – digital signature algorithm, ANSI X9.30-1 standard;
[2] RSA signature algorithm, ANSI X9.31-1 standard;
[3] ECDSA – elliptic curve digital signature algorithm, IEEE P1363/D2 (1998);
[4] **B. Hołyst**: Kryminalistyka, wyd. XI, LexisNexis, Warszawa 2007, s. 805;
[5] **B. Hołyst, J. Pomykała:** Podpis elektroniczny – aspekty kryminalistyczne, prawne i informatyczne, „Problemy Kryminalistyki" 2007, nr 256, s. 5–22;
[6] **H. Delfs, H. Knebl:** Introduction to cryptography, Springer 2002, s. 221–224;
[7] **B. Hołyst, J. Pomykała:** op.cit., s. 5–22;
[8] **A. Shamir:** How to share a secret, „Communications of the ACM" 1979, nr 22 (1), s. 612–613;
[9] **R. Di Pietro, L.V. Mancini, G. Zanin:** Efficient and adaptive threshold signatures for ad hoc networks, „Electronic Notes in Theoretical Computer Science" 2007, nr 171, s. 93–105;
[10] **J. Pomykała, S. Barabasz:** Elliptic curve based threshold Proxy signature scheme with known signers, „Fundamenta Informaticae" 2006, nr 69 (4), s. 411–425;
[11] **J. Pomykała:** On hierarchical structures and access control, Proceedings of 5th International Conference APLIMAT 2006, Bratislava, s. 135–145;
[12] **J. Pomykała, T. Warchoł:** Threshold signatures in dynamic groups, Proceedings of Future Generation Communication and Networking 2007, IEEE Computer Science, s. 32–37.
[13] **Y. Desmedt:** Society and group oriented cryptography: A new concept, CRYPTO '87, LNCS 293, 1988, s. 120–127;
[14] **R. Gennaro, S. Jarecki, H. Krawczyk, T. Rabin:** Robust threshold DSS signatures, Eurocrypt '96, LNCS 1070, 1996, s. 354–371;
[15] **J. Pomykała, T. Warchoł:** op.cit., s. 5–22;
[16] **R. Di Pietro, L.V. Mancini, G. Zanin:** op.cit., s. 93–105;
[17] **B. Przydatek, D. Song, A. Perrig:** SIA: Secure information aggregation in sensor networks, reprint 2003;
[18] **H. Ghodosi, J. Pieprzyk, R. Safavi–Naini:** Dynamic threshold cryptosystems: a new scheme in group oriented cryptography, Proceedings of PRAGOCRYPT '96, s. 370–379;
[19] **Y. Desmedt:** Threshold cryptography, „European Transactions on Telecommunications" 1994, nr 5 (4), s. 449–457;
[20] **B. Nakielski, J. Pomykała, J.A. Pomykała:** A multi-threshold signature ukaże się w Journal of Telecommunication and Information Technology 2008 (1);
[21] **R. Di Pietro, L.V. Mancini, G. Zanin:** op.cit., s. 93–105;
[22] **A. Boldyreva:** Threshold signatures, multisignatures and blind signatures based on the gap Diffie-Hellman group signature scheme, LNCS 2567, 2003, s. 31–44;
[23] **J. Pomykała, T. Warchoł:** op.cit., s. 5–22;
[24] **R. Di Pietro, L.V. Mancini, G. Zanin:** op.cit., s. 93–105;
[25] **B. Hołyst, J. Pomykała:** op.cit., s. 5–22;
[26] Ibidem;
[27] **V. Shoup:** Practical threshold signatures, Eurocrypt LNCS 1807, 2000, s. 207–220;
[28] **R. Rivest, A. Shamir, L.M. Adleman:** A method for obtaining digital signatures and public key cryptosystems, Communications of the ACM 1978, nr 21 (2), 120–126;
[29] **J. Pomykała, J.A. Pomykała:** Systemy informacyjne. Modelowanie i wybrane techniki kryptograficzne, MIKOM, Warszawa 1999; s. 149–163;
[30] **J. Pomykała, T. Warchoł:** op.cit.;
[31] **A. Boldyreva:** op.cit.

e-mail: clkpk@policja.gov.pl