Narodowy Fundusz Zdrowia

Projekt: *Usługa doradztwa eksperckiego w ramach programu wdrożenia silnych mechanizmów identyfikacji i uwierzytelniania na potrzeby systemu Rejestru Usług Medycznych II (RUM II) dla Centrali Narodowego Funduszu Zdrowia*

Analiza przyjętych rozwiązań technologicznych Projektu RUM II

Wersja: *3.2*
Autor: *Zespół Wykonawcy*
Data: *2014-11-03*

4pi Sp. z o.o. 03-924 Warszawa, ul. Niekłańska 27/5, tel/fax (22) 616 33 42, www.4pi.pl

ENIGMA Systemy Ochrony Informacji Sp. z o. o., ul. Jutrzenki 116, 02-230 Warszawa, tel. 22 570 57 10, www.enigma.com.pl

SPIS TREŚCI

SPIS TABEL

SŁOWNIK

Patrz dokument „Finalna koncepcja rozwiązań w zakresie Projektu RUM II”.

1. ANALIZA MECHANIZMÓW KRYPTOGRAFICZNYCH

Rozdział zawiera podsumowanie analizy mechanizmów kryptograficznych, która znajduje się w Dodatku A niniejszego opracowania.

Przyjęte w założeniach do projektu podejście oparte jest na zachowaniu możliwie dużej zgodności z normami i standardami, co oznacza redukcję ryzyka związanego z zastosowaniem mechanizmów, których siła nie została w wystarczającym stopniu zweryfikowana. W powiązaniu z wnioskami z rozdziału 2 należy uznać, że zastosowanie algorytmów dopuszczonych do użytkowania na podstawie standardu ETSI TS 102 176-1 jest wystarczająco bezpieczne w perspektywie eksploatacji systemu.

Dopuszcza się stosowanie algorytmów:

- RSA lub DSA o długości kluczy nie mniejszej niż 2048 bitów do podpisu i uwierzytelnienia, albo ECDSA o długości krzywej nie krótszej niż 256 bitów (przy czym do podpisu i do uwierzytelnienia należy używać dwóch różnych par kluczy).
- AES o długości klucza co najmniej 128 bitów albo 3DES z trzema niezależnymi kluczami do szyfrowania danych,
- SHA-2 (co najmniej SHA-256), jako funkcję skrótu wykorzystywaną w podpisach elektronicznych.

Uwaga!

1. Dopuszcza się stosowanie SHA-1 wyłącznie do celów zapewnienia wstecznej kompatybilności (w zakresie wyliczania skrótów służących do weryfikacji integralności, a nie do podpisów elektronicznych) z zaleceniem wycofywania się z używania tego algorytmu.
2. Zaleca się implementację algorytmu SHA-3 w celu zwiększenia bezpieczeństwa systemu w perspektywie jego długoterminowego używania i zastosowanie tego nowego algorytmu, w przypadku pojawienia się pierwszych sygnałów o wykryciu słabości w SHA-2.
3. Dla urzędów certyfikacji kluczy i dla kluczy infrastruktury sugeruje się stosowanie algorytmów o wyższym poziomie bezpieczeństwa niż dla użytkowników.
4. Należy zwrócić uwagę na kwestie związane z ochroną patentową wielu implementacji algorytmów opartych na krzywych eliptycznych (są to m.in. patenty firm Certicom, Hewlett-Packard, RSA Data Security oraz National Security Agency).
5. W celu zwiększenia szybkości działania przy podpisie i weryfikacji podpisu (np. w celu uniknięcia wąskich gardeł przy weryfikacji podpisów pod danymi sprawozdawczymi) sugeruje się użycie krzywych eliptycznych jako algorytmu podpisu.

Autorzy niniejszego opracowania, biorąc pod uwagę względy bezpieczeństwa systemu i względy wydajnościowe zalecają, aby w początkowym okresie eksploatacji systemu wybrać następujące parametry algorytmów kryptograficznych:

1. W infrastrukturze klucza publicznego urzędy certyfikacji (RootCA i CA dla certyfikatów X.509 oraz serwery usług OCSP i DAT) powinny wykorzystywać klucze RSA o długości 3072 bity.

2. Klucze użytkowników końcowych zarówno kart KUZ, KSA, jak i ew. kart KSM powinny mieć długość 2048 bitów i mieć mały wykładnik publiczny – obecnie co najmniej postaci F_4 (ze względu na szybszą weryfikację podpisu RSA z kluczami o długości 2048 bitów i o małym wykładniku publicznym w stosunku do weryfikacji podpisu na krzywej eliptycznej o długości 256 bitów).
3. Klucze infrastruktury (określane jako klucze kontrolne w CWA 14167-1), czyli klucze operatorów i administratorów powinny być kluczami RSA o długości 2048 bitów i być przechowywane na kartach elektronicznych.
4. Stosowana funkcja skrótu powinna należeć do rodziny SHA-II o bloku co najmniej 256 bitów.
5. Zaleca się zaimplementowanie w rozwiązaniu funkcji SHA-III.
6. Do celów szyfrowania powinien być stosowany algorytm AES o długości klucza co najmniej 128 bitów.
7. Dla urzędu certyfikacji CV certyfikatów może być stosowany algorytm RSA o długości kluczy 2048 bitów.
8. Jako algorytm uwierzytelnienia między kartami (KSM i KUZ) może być stosowany algorytm RSA o długości kluczy 2048 bitów.
9. Do celów wymiany informacji np. pomiędzy urzędem certyfikacji, a punktem rejestracji (nawet jeśli będzie on automatyczny) powinny być stosowane RSA o długości kluczy 2048 bitów, albo algorytm Diffie-Helmana o |p| = 2048 bity i |q| = 224 lub 256 bitów, albo ECDH zgodnie z tabelą 5 NIST SP 800-131A.

Ze względu na to, że czas weryfikacji podpisu przy zwiększeniu długości klucza RSA i odpowiednio długości krzywej eliptycznej przestaje być czynnikiem dominującym, należy przeprowadzić testy systemów również dla krzywych eliptycznych o długości 372 bity, aby, przy konieczności zmiany długości kluczy w systemie, była możliwość przejścia na krzywe eliptyczne, jako algorytm bezpieczniejszy. Należy też pamiętać o wymaganiu, by aplikacje podpisujące i weryfikujące zapewniały możliwość obsługi zarówno RSA, jak i krzywych eliptycznych oraz aby przeprowadzić testy infrastruktury PKI w oparciu o krzywe eliptyczne.

Zagadnienie związane z wyborem krzywej na której odbywać się będą operacje kryptograficzne należy ustawić w ramach dialogu technicznego z dostawcami kart.
W ramach dialogu technicznego należy zweryfikować również możliwość stosowania krzywych eliptycznych w CV certyfikatach byłoby to korzystne z uwagi na mniejszą ilość miejsca zajmowanego na karcie i ograniczenie wydajności kart. Wówczas należy zapewnić również możliwość wydawania certyfikatów na krzywych eliptycznych w gałęzi CV certyfikatów w infrastrukturze PKI.

2. OCZEKIWANE OGRANICZENIA CZASOWE BEZPIECZEŃSTWA ALGORYTMÓW

Rozdział podsumowuje ograniczenia czasowe bezpieczeństwa algorytmów, których pełna analiza znajduje się w Dodatku A niniejszego opracowania.

Bezpieczeństwo algorytmów stosowanych w ramach RUM II należy rozpatrywać w dwóch kontekstach:

1) bezpieczeństwa algorytmów zaimplementowanych na kartach KUZ i KSA oraz
2) bezpieczeństwa algorytmów zaimplementowanych w oprogramowaniu systemów informatycznych NFZ.

Podział ten jest o tyle istotny, że w miarę upływu czasu (i potencjalnego wykrywania słabości w zastosowanych algorytmach), koszt ich wymiany w ramach oprogramowania systemów informatycznych jest znacząco niższy niż w przypadku kart. Przyjmując 10-letni okres ważności karty należy wybrać takie algorytmy i takie długości kluczy, aby z dużą dozą ufności można było przyjąć zachowanie ich bezpieczeństwa w tej skali czasu. Z punktu widzenia bezpieczeństwa systemu RUM II decydującą rolę odgrywa bezpieczeństwo algorytmu składania podpisu. Biorąc pod uwagę dostępność kart należy stwierdzić, że zastosowanie algorytmu RSA o długości klucza 2048 bitów jest wystarczające przy założeniu 10-letniego okresu eksploatacji kart. Jednak zalecane użycie algorytmów do konkretnych zastosowań (np. do składania podpisu) nie dopuszcza ich stosowania przez tak długi okres czasu. Czyli dopuszczalne wykorzystanie kluczy o określonej długości, na ustalony okres czasu, nie implikuje wykorzystania jednego klucza (bez zmieniania go na inny) przez cały ten okres czasu.

Zgodnie z zaleceniami organizacji normalizacyjnych i innych podmiotów, klucze do podpisów nie powinny być wykorzystywane dłużej niż 4 lata (NIST proponuje 1-3 lat, NATO – 1,5 roku, normy UE 2-4 lat, polska ustawa o podpisie elektronicznym – dla kwalifikowanych certyfikatów 2 lata, w przyszłości 5 lat). Biorąc pod uwagę inne aspekty, w systemie NFZ klucze RSA o długości 2048 bitów - dla rozwiązań innych niż „podpis medyczny” - nie powinny być wykorzystywane przez okres dłuższy niż 5 lat (z marginesem na wymianę). W przypadku konieczności wydłużenia okresu życia kluczy należy zdecydować o zmianie algorytmu na krzywe eliptyczne i dokonać ponownej analizy.

Należy zauważyć, że powyższe okresy ważności kluczy i certyfikatów dotyczą kluczy jednostek końcowych: Subskrybentów lub urządzeń. Okresy ważności kluczy urzędów certyfikacji i usług znakowania czasem lub OCSP są zwykle dłuższe. NATO rekomenduje 6-letni okres ważności klucza dla urzędu certyfikacji najwyższego poziomu (RootCA), przy kluczach RSA o długości 4096 bitów lub 2048 bitów, ale jednocześnie wycofuje się ze stosowania RSA w 2015 roku. Dalej dopuszcza stosowanie ECDSA o długości kluczy 384 bity przez okres 12 lat dla urzędów certyfikacji najwyższego poziomu, przez okres 3 lat dla operacyjnych urzędów certyfikacji, które wydają klucze użytkownikom końcowym i dla Punktów Rejestracji.

Na bezpieczeństwo karty składa się również odporność algorytmów symetrycznych 3DES i AES. Na bazie obecnego stanu wiedzy należy stwierdzić, że algorytmy te mogą być bezpiecznie eksploatowane co najmniej przez założony okres.

3. ANALIZA PODATNOŚCI

3.1. Wpływ wprowadzenia RUM II na potencjalne nadużycia

Tabela poniżej zawiera zestawienie potencjalnych rodzajów nadużyć wraz z oceną wpływu systemu RUM II

LP	Rodzaj nadużycia	Wpływ RUM II
1.	Fikcyjne świadczenia zdrowotne realizowane w placówkach służby zdrowia	Redukcja[1]
2.	Fikcyjne świadczenia zdrowotne realizowane poza placówkami służby zdrowia	Brak lub niewielka redukcja
3.	Realizacja świadczeń niezgodnie z kontraktem (np. zbyt mała liczność grupy, niewłaściwy czas trwania świadczenia, wydawanie leków lub zleceń przez nieuprawniony personel)	Redukcja dla niektórych typów świadczeń
4.	Realizacja świadczeń przez personel niezgodny z zadeklarowanym potencjałem	Redukcja w II etapie projektu
5.	Realizacja świadczeń niezgodnie z harmonogramem	Redukcja w II etapie projektu
6.	Fikcyjnie wydłużony czas wizyty (dla wizyt opłacanych proporcjonalnie do czasu trwania), zwłaszcza u psychiatrów i psychologów	Redukcja
7.	Fikcyjni pacjenci	Redukcja
8.	Fikcyjni świadczeniodawcy	Redukcja
9.	Nadmiarowe świadczenia zdrowotne	Brak
10.	Zlecanie świadczeń (np. badań), często nadmiarowych, w instytucjach, w których zlecający mają udziały	Brak
11.	Martwe dusze na listach pacjentów rozliczanych stawką kawitacyjną	Redukcja (wybór lekarza potwierdzany kartą KUZ)[2]
12.	Świadczenia na rzecz nieubezpieczonych pacjentów	Niewielka redukcja – o ile do identyfikacji będzie służyła karta i dokument ze zdjęciem
13.	Wystawianie recept refundowanych dla pacjentów bez ubezpieczenia z wykorzystaniem danych pacjentów ubezpieczonych	Redukcja

[1] Przy dystrybucji kart KUZ za pośrednictwem POZ – nadal istnieje możliwość wykorzystania nieodebranych kart KUZ do sprawozdawania fikcyjnych świadczeń

[2] Przy dystrybucji kart KUZ za pośrednictwem POZ – nadal istnieje możliwość wykorzystania nieodebranych kart KUZ do sprawozdawania fikcyjnych pacjentów

14.	Modyfikacja wystawionych recept przez pacjentów	Brak (P1)
15.	Zmowy lekarz – farmaceuta lub lekarz – pacjent – farmaceuta w celu popełniania nadużyć w obrocie lekami	Brak
16.	Nadużycia w obrocie lekami w leczeniu szpitalnym	Brak
17.	Zmowy lekarz-pacjent w lecznictwie uzdrowiskowym	Niewielka redukcja
18.	Fikcyjne świadczenia w leczeniu uzdrowiskowym	Redukcja
19.	Nieuzasadnione zwolnienia lekarskie	Brak
20.	Fałszowanie (przedłużanie) zwolnień przez pacjentów	Redukcja (podpisy kwalifikowane lekarza od 2017 r.)
21.	Podwójne finansowanie świadczeń (za odpłatnością i jednoczesnym rozliczaniem ze środków publicznych)	Brak
22.	Stosowanie tanich zamienników medycznych, a pobieranie pełnej stawki refundacji	Brak
23.	Wielokrotne wyłudzanie przez pacjentów od różnych lekarzy recept na refundowane leki podlegające kontroli obrotu	P1
24.	Odbiór zleceń dla osób nieżyjących	Brak
25.	Zamiana kilku refundowanych zleceń na jedno droższe	Brak
26.		

3.2. Nowe podatności wynikające z informatyzacji procesów, zwiększenia funkcjonalności systemu i zastosowania kart

1.	Nieprawidłowa personalizacja karty KUZ lub wykorzystanie błędnych danych do personalizacji (personalizacja graficzna nie odpowiada personalizacji w warstwie elektronicznej)
2.	Nieprawidłowa personalizacja karty KSM lub wykorzystanie błędnych danych do personalizacji (personalizacja graficzna nie odpowiada personalizacji w warstwie elektronicznej)
3.	Nieprawidłowa personalizacja karty KSA lub wykorzystanie błędnych danych do personalizacji (personalizacja graficzna nie odpowiada personalizacji w warstwie elektronicznej)
4.	Wydanie karty KUZ osobie innej niż właściciel
5.	Wydanie karty KSM osobie innej niż właściciel
6.	Wydanie karty KSA osobie innej niż właściciel
7.	Kradzież/wyłudzenie karty KUZ
8.	Kradzież/wyłudzenie karty KSM

9.	Kradzież/wyłudzenie karty KSA
10.	Chwilowe wypożyczenie i wykorzystanie karty KUZ
11.	Chwilowe wypożyczenie i wykorzystanie karty KSM (razem z PIN'ami)
12.	Chwilowe wypożyczenie i wykorzystanie karty KSA (razem z PIN'ami)
13.	Klonowanie recept elektronicznych zapisanych na karcie (o ile takie użycie kart będzie stosowane)
14.	Nieuprawniony dostęp farmaceuty do starych recept pacjent (o ile takie użycie kart będzie stosowane)
15.	Nieuprawniony dostęp do danych pacjenta na karcie lub w systemie informatycznym (za pomocą karty KUZ pacjenta – skradzionej/wyłudzonej/wypożyczonej)
16.	Nieuprawniony dostęp do danych pacjenta (za pomocą karty KSM – skradzionej/wyłudzonej/wypożyczonej) lub karty KSM, która powinna zostać zwrócona, ze względu na zawieszenie/wygaśnięcie prawa wykonywania zawodu
17.	Przełamanie zabezpieczeń karty KUZ
18.	Przełamanie zabezpieczeń karty KSM
19.	Przełamanie zabezpieczeń karty KSA
20.	Naruszenie integralności danych w systemach informatycznych
21.	Niedostępność systemu informatycznego CSIOZ P1
22.	Niedostępność systemu informatycznego NFZ
23.	Niedostępność innych systemów
24.	Brak dostępu do Internetu (w świadczeniodawcy/ w aptece/ w miejscu odbioru Zleceń)
25.	Awaria komputera
26.	Brak zasilania
27.	Wyciek danych z systemu informatycznego
28.	Wyciek danych z innych nośników (np. z kopii zapasowych)
29.	Wyciek danych z systemu SZUK(np. przekazywanych do personalizacji)
30.	Wyciek danych podczas procesu niszczenia danych (jeśli karty niszczy NFZ)
31.	Błędne przypisanie praw dostępu w systemie informatycznym
32.	Brak ubezpieczenia na rzecz dochodzenia roszczeń przez użytkowników w systemie RUM II (np. z powodu naruszenia dóbr osobistych)

4. ANALIZA KONCEPCJI PIECZĘCI ELEKTRONICZNEJ KUZ

Wg Koncepcji w Fazie II zdarzenia medyczne raportowane do NFZ będą opatrywane – przy pomocy karty KUZ – „pieczęcią medyczną", natomiast w fazie I pieczęć ta będzie wykorzystywana tylko do potwierdzenia faktu przybycia świadczeniobiorcy do świadczeniodawcy, stąd pieczęć będzie składana tylko na etapie rejestracji pacjenta.

Wg eIDAS rozróżnienie między pieczęcią a podpisem elektronicznym nie dotyczy aspektu technicznego – w obu przypadkach (złożenie pieczęci lub podpisu elektronicznego) odbywa się przez wykonanie analogicznych przekształceń matematycznych za pomocą karty elektronicznej. Różnica dotyczy aspektu formalno-proceduralnego – w przypadku bowiem podpisów elektronicznych w stosownym certyfikacie w polu „subject" byłoby zapisane imię, nazwisko i PESEL świadczeniobiorcy, natomiast w przypadku pieczęci elektronicznej byłyby to nazwa, REGON itp. atrybuty osoby prawnej.

Aktualne regulacje prawne, zarówno unijne, jak i polskie, nie definiują pieczęci elektronicznej. Pojęcie to pojawia się dopiero w rozporządzeniu eIDAS, które zacznie obowiązywać w obszarze tzw. Trust Services, obejmujących podpis i pieczęć elektroniczną, dopiero od połowy 2016 r. Analiza stosownych zapisów rozporządzenia UE, mimo braku aktów wykonawczych, skłania do wniosku, że pieczęć elektroniczna dotyczy przede wszystkim uwierzytelniania dokumentów wydanych przez osobę prawną, jak również do uwierzytelnienia wszelkich zasobów cyfrowych osoby prawnej, takich jak kod oprogramowania lub serwery – patrz pkt 65 preambuły. W związku z tym używanie karty KUZ w celu potwierdzania faktu fizycznego przybycia do placówki świadczeniodawcy (Faza I) i/lub potwierdzania udzielenia świadczenia medycznego (Faza II), to zdecydowanie materia podpisów elektronicznych świadczeniobiorcy, który w tym zakresie (i tylko w tym) działa jako pacjent, a nie jako osoba prawna (NFZ).

W związku z powyższym proponuje się zmienić nazewnictwo i stosować zamiast „pieczęci medycznej" pojęcie „elektronicznego podpisu medycznego", w skrócie „podpisu medycznego". Odróżnienie podpisu medycznego od innych postaci podpisów elektronicznych, w tym w szczególności „bezpiecznego" (kwalifikowanego) podpisu elektronicznego, czy podpisu cyfrowego (uwierzytelnienie on-line i podpisywanie losowych wyzwań), jest o tyle niezbędne, że w odróżnieniu od tych innych postaci, „podpis medyczny" nie będzie wymagał odblokowania karty KUZ przy pomocy PIN-u. W związku z tym należy wyraźnie ograniczyć jego zastosowanie tylko i wyłącznie do ww celów związanych z refundowaniem przez NFZ udzielonych świadczeń medycznych oraz dla celu potwierdzania autentyczności samej karty. Żadne inne zastosowania nie wchodzą w rachubę, gdyż brak PIN-u narażałby użytkownika karty na nieakceptowane ryzyko kradzieży tożsamości i wykorzystania jej w różnych innych przypadkach.

Zgodnie ze standardami dot. PKI ograniczenia zakresu stosowania danych podpisów elektronicznych dokonuje się poprzez odpowiednie zapisy w polityce i regulaminie certyfikacji. Stąd NFZ, wystawiając certyfikaty podpisu medycznego, będzie musiał uprzednio zdefiniować treść polityki i regulaminu certyfikacji, opublikować je na stronach www oraz – rekomendujemy – uzyskać dla tej polityki odpowiedni identyfikator (ang. OID) i zarejestrować ten identyfikator w Krajowym Rejestrze Identyfikatorów Obiektów

prowadzonym w imieniu Polskiego Komitetu Normalizacyjnego przez Unizeto Technologies SA – https://www.krio.pl/krio/main.xml.

Można spotkać interpretację, zgodnie z którą „podpis elektroniczny" wymaga PIN'u, natomiast „pieczęć elektroniczna" – skoro jest anonimowa – to będzie stosowana przez różne (chociaż zapewne upoważnione) osoby, które będą ją składały jako „osoba prawna" i dlatego jej zastosowanie nie będzie związane z PIN-em. Albo argument, że „pieczęć elektroniczna" będzie rozwiązaniem do masowego stemplowania faktur elektronicznych, gdzie podawanie PIN-u jest niemożliwe z powodu ograniczeń technicznych (supermarket wystawiający wiele faktur na sekundę). Autorzy takich interpretacji mylnie utożsamiają „PIN odblokowujący" z koniecznością jego podawania przy każdym składaniu podpisu elektronicznego. Regulacje prawne, np. § 7 ust. 8 polskiego rozporządzenia w sprawie określenia warunków technicznych i organizacyjnych dla kwalifikowanych podmiotów , pozwalają na tryb „jeden PIN – wiele podpisów", ale musi to być świadoma decyzja podpisującego . Również uważna lektura treści rozporządzenia eIDAS skłania do wniosku, że nie należy oczekiwać żadnych różnic w kontekście PIN-u między „podpisem" i „pieczęcią" elektroniczną. Kluczowy jest tu bowiem art. 39 rozporządzenia eIDAS który wskazuje, że takie same wymagania dla bezpiecznych urządzeń będą dotyczyły „pieczęci elektronicznej" i „podpisu elektronicznego":

Artykuł 39

Kwalifikowane urządzenia do składania pieczęci elektronicznej

1. Art. 29 stosuje się odpowiednio do wymogów dotyczących kwalifikowanych urządzeń do składania pieczęci elektronicznej.

2. Art. 30 stosuje się odpowiednio do certyfikacji kwalifikowanych urządzeń do składania pieczęci elektronicznej.

Zwracamy jednocześnie uwagę, że brak PIN-u przy podpisie medycznym stoi w sprzeczności z aktualnymi wymaganiami (i zapewne przyszłymi, w tym aktami wykonawczymi do eIDAS) dot. tzw. bezpiecznych urządzeń do składania podpisu elektronicznego. Jednak nie stanowi to najmniejszej przeszkody dla wymagania posiadania dla warstwy elektronicznej karty KUZ certyfikatu SSCD. Otóż innym wymaganiom podlega podpis medyczny, a innym podpis cyfrowy również możliwy do złożenia przy pomocy karty KUZ. Podpis medyczny będzie mechanizmem potwierdzania autentyczności pewnych zdarzeń związanych z refundacją świadczeń medycznych przez polski NFZ. W związku z czym taki podpis będzie wyłączony z zakresu rozporządzenia eIDAS, które nie dotyczy systemów zamkniętych, i tym samym całkowicie pozbawionych aspektów *transgraniczności*".

Reasumując: w stosownym certyfikacie podpisu medycznego należy umieścić w polu subject imię, nazwisko i PESEL świadczeniobiorcy. W standardowym rozszerzeniu „Polityka certyfikacji" (ang. certificatePolicies) należy wskazać politykę, której treść określi zakres stosowania podpisu medycznego – świadczenia związane z ochroną zdrowia i potwierdzanie autentyczności karty KUZ, czyli w rozszerzeniu keyUsage należy ustawić odpowiednio dwa bity: „contentCommitment" i „digitalSignature". Ten pierwszy będzie związany z podpisywaniem „daty" wykonania świadczenia lub pobytu w placówce świadczeniodawcy, a ten drugi dotyczy podpisywania losowych wyzwań w ramach protokołu challenge-response przy potwierdzaniu autentyczności karty KUZ (udowodnienia posiadania stosownego klucza

prywatnego, komplementarnego z publicznym, zawartym w certyfikacie wystawionym przez CA NFZ).

5. ANALIZA INTERFEJSU STYKOWEGO KARTY

W przypadku kart elektronicznych formatu ID-1 wchodzą w rachubę dwa interfejsy warstwy elektronicznej: stykowy i bezstykowy („radiowy"). Oba warianty[3] mają swoje wady i zalety, przy czym w przypadku karty KUZ rekomendujemy użycie interfejsu stykowego. Przemawiają za tym następujące argumenty:

a) w przypadku interfejsu radiowego bezwzględnie należy zadbać o to, aby obywatel nie był narażony na „wyłudzenie podpisu" przy pomocy karty znajdującej się w zakrytej kieszeni. Interfejs stykowy niejako *z definicji* eliminuje takie zagrożenie, natomiast w przypadku interfejsu bezstykowego wchodzą w rachubę następujące rozwiązania: czytnik ze skanerem pola MRTD (tak jak w paszportach) lub protokół PACE[4], gdzie obywatel dodatkowo podaje PIN. Rozwiązanie „paszportowe" jest powszechnie stosowane na świecie, ale cena czytnika zaczyna się od kwoty 400 zł, natomiast założenie zastosowania PACE stoi w sprzeczności z ideą „automatycznej" (bez PIN'u) autoryzacji udzielanego świadczenia medycznego. PACE przewiduje też wsparcie dla mechanizmu generowania jednorazowego kodu przez kartę elektroniczną i jego wyświetlenia na specjalnym wyświetlaczu wbudowanym do karty. Rozwiązanie być może przyszłościowe, ale aktualnie nie wyszło poza etap prototypów i należy mieć ogromne obawy co do 10-letniej trwałości wyświetlacza wbudowanego do karty KUZ, nie mówiąc o dodatkowych ograniczeniach związanych z layout'em takiej karty KUZ;
b) użycie interfejsu stykowego silnie rekomenduje dokument CWA 15974:2009 „Interoperability of the electronic European Health Insurance Cards (WS/eEHIC)", który opisuje elektroniczną europejską kartę ubezpieczenia zdrowotnego – np. w pkt. 0.2 „Wprowadzenia" (ang. *Introduction*):

   > *For data storage interoperability (read/write/update of eEHIC data) it is strongly recommended that the card edge should conform to the CEN/TS 15480-2 specification at least with contact physical interface.*

   Dokument CWA 15974 nie został formalnie zaaprobowany przez wszystkie państwa członkowskie UE (stąd nie ma charakteru obligatoryjnego), jednak z drugiej strony trzeba zaznaczyć, że jego treść powstawała przy udziale szeregu ekspertów z różnych krajów UE;
c) zastosowanie konkretnego interfejsu karty KUZ determinuje odpowiednie wymagania dla czytników, które są wielokrotnie droższe dla wersji bezstykowej niż dla stykowej.

Ewentualne argumenty przemawiające na korzyść interfejsu bezstykowego są następujące:

a) wymaganie udzielenia gwarancji na 10-letni okres używania karty KUZ jest łatwiejsze do uzyskania w przypadku interfejsu radiowego, niż stykowego – odpada problem zużywania się styków w przypadku współpracy karty z czytnikami różnej kategorii jakości. Jeszcze kilka lat temu wiodący producenci kart nie godzili się na 10-letnią gwarancję w przypadku karty stykowej, ale aktualnie uległo to zmianie – można uzyskać taką gwarancję;

[3] można rozważać jeszcze rozwiązanie „dualne", w którym jeden procesor warstwy elektronicznej ma możliwość wymiany informacji z otoczeniem przez oba interfejsy

[4] ang. *Password Authenticated Connection Establishment*

b) ewentualne połączenie karty KUZ z dokumentem, który miałby funkcjonalność „dokumentu podróży” (biometryczny dowód osobisty lub paszport), wymagałoby interfejsu bezstykowego, jednak nie przewiduje się takiej funkcjonalności w przypadku karty KUZ.

6. ANALIZA WYKONANIA I CERTYFIKACJI SPECJALIZOWANEGO APLETU JAVA

Zgodnie z założeniami KUZ karta musi mieć możliwość realizacji w przyszłości „dodatkowej funkcjonalności" (np. identyfikacja biometryczna *match on card*). Istnienie mechanizmu pozwalającego na wprowadzenie dodatkowej struktury danych z odpowiednimi prawami dostępu, w oparciu o istniejące mechanizmy systemu operacyjnego, nie jest wystarczające. W związku tym zapewnienie możliwości wprowadzenia w przyszłości „nowej funkcjonalności" determinuje użycie kart typu „Java" i wyklucza stosowanie kart „natywnych".

Zwracamy jednocześnie uwagę, że wybór dwóch różnych dostawców kart drastycznie ogranicza scenariusz wprowadzenia „nowej funkcjonalności" w obu kartach, gdyż nie można wykluczyć, że dodatkowa funkcjonalność będzie możliwa do implementacji tylko na karcie jednego dostawcy.

Wprowadzenie nowych apletów do systemów operacyjnych kart wiąże się również z problemem braku specyfikacji normalizujących etap „prepersonalizacji" karty. Otóż normy i standardy dotyczące kart elektronicznych pomijają aspekt inicjacji karty, który jest specyficzny dla każdego producenta kart elektronicznych i jest przedmiotem indywidualnych uzgodnień między dostawcą i użytkownikiem. Z drugiej strony wiodący dostawcy kart elektronicznych zgodnych z SSCD stosują dość ograniczony zestaw mechanizmów wymaganych do przygotowania karty do etapu personalizacji. Ze względu na to, że zakładamy, że wykonawcą systemu SZUK, czyli *integratorem* aplikacji NFZ i podmiotów personalizujących, będzie jeden z dostawców kart, to nie należy się spodziewać, że wykonawca systemu SZUK napotka jakieś olbrzymie przeszkody w stosunku do „drugiej" karty. Jednak problem osadzania w przyszłości nowych apletów w kartach KUZ nie będzie mógł być jednoznacznie zdefiniowany na etapie SIWZ (wspomniany brak standaryzacji) – jedynie może być zapisana konieczność obsługi takiej funkcji w przyszłości przez wykonawcę systemu SZUK.

Podsumowując: zastosowanie certyfikowanej karty Java zgodnej z profilem „SSCD" zawartym w standardzie CWA 14169 Secure Signature-Creation Devices "EAL 4+", z certyfikatem wydanym przez ciało wyznaczone (ang. *designated body*) w rozumieniu dyrektywy Parlamentu Europejskiego i Rady 1999/93/WE z dnia 13 grudnia 1999 r. w sprawie wspólnotowych ram w zakresie podpisów elektronicznych, przy pozostawieniu w rękach NFZ (funkcjonalność systemu SZUK'a) kontroli nad możliwością osadzania innych apletów na karcie, wydaje się całkowicie wystarczające do realizacji wszystkich założonych celów dla kart KUZ, KSA i KSM.

7. ANALIZA POZIOMU ZABEZPIECZEŃ FIZYCZNYCH KARTY

10-letni okres trwałości kart KUZ wyklucza szereg technologii stosowanych w kartach bankowych, które mają zdecydowanie krótsze czasy eksploatacji, rzędu 2-3 lat (w pojedynczych przypadkach 4-5 lat). W związku z tym rekomenduje się zastosowanie technologii poliwęglanowej. Oto argumenty przemawiające za takim rozwiązaniem:

a) technologia poliwęglanowa jest oferowana przez wszystkich wiodących dostawców kart;
b) poliwęglan jest jedynym materiałem, który został praktycznie pozytywnie zweryfikowany w kontekście 10-letniej trwałości w Polsce (dowody osobiste);
c) nowe wymagania na polskie prawo jazdy zawarte w rozporządzeniu Ministra Transportu przewidują m.in. „*Prawo jazdy ma formę karty zbudowanej z wielowarstwowego poliwęglanu, przy czym warstwy zewnętrzne są przezroczyste, a warstwy środkowe nieprzezroczyste. Na karcie nadrukowano technikami offsetową i sitodrukową elementy graficzne i zabezpieczające, a techniką druku inkjetowego naniesiono kolorową fotografię posiadacza prawa jazdy*”, a taki dokument będzie z pewnością tak samo przechowywany przez użytkownika jak karta KUZ;
d) wprowadzenie takiego wymagania nie będzie budziło wątpliwości (prawo jazdy, dowód osobisty) i pozwoli na istotne zminimalizowanie ryzyka, że pojawią się dwa rodzaje kart „lepsze i gorsze”, co mogłoby mieć miejsce, gdyby wymaganie pozwalało na wybór również innej niż poliwęglan technologii.

Wiodący dostawcy kart identyfikacyjnych i bankowych oferują szereg zabezpieczeń fizycznych w swoich kartach. Implementacja niektórych z nich jest związana z dodatkową opłatą, ale jest całe spektrum zabezpieczeń, które są niejako „w cenie”, czyli ich zastosowanie lub rezygnacja nie mają wpływu na cenę karty, również w przypadku karty poliwęglanowej. W związku z tym uważamy za zasadne wprowadzić do specyfikacji technicznej karty KUZ (również KSA i KSM) kilka „bezkosztowych” zabezpieczeń w warstwie graficznej. Stwierdzamy, że wartym rozważenia jest zestaw zabezpieczeń, o którym mowa w regulacji unijnej dot. wzorów poświadczeń posiadanych uprawnień do przewozu materiałów niebezpiecznych[5]. Zapisy umowy europejskiej stanowią, że „*Zaświadczenie powinno wykonane z tworzywa sztucznego, a jego wymiary powinny być zgodne z ISO 7810:2003 ID-1. Zaświadczenie powinno być koloru białego, a litery czarne. Zaświadczenie powinno zawierać dodatkowe zabezpieczenie, takie jak hologram, druk UV lub tło giloszowe.*”.

Zastosowane zabezpieczenia w przypadku polskiego zaświadczenia ADR są zobrazowane tutaj: http://info-car.pl/infocar/adr/poznaj-dokument-adr.html

Poniżej przedstawiamy zakres rekomendowanych zabezpieczeń graficznych karty KUZ, KSA i KSM, których zastosowanie nie będzie miało istotnego wpływu na koszt personalizacji:

a) gilosz – zachodzące na siebie tła różnych pól (ciągłe cienkie linie i pola są tworzone z wykorzystaniem wielu kolorów w taki sposób, że staje się niemożliwe skopiowanie

[5] Umowa europejska dotycząca międzynarodowego przewozu drogowego towarów niebezpiecznych obowiązująca od dnia 1 stycznia 2013 r. (Dziennik Ustaw z 2012 r. poz. 815)

takich wzorów za pomocą powszechnie stosowanych urządzeń kopiujących. Takie „zwykłe" urządzenia, włączając w to fotokopiarki, wykonują druk kolorowy w postaci różnokolorowych pixeli (małych kropek), co jest łatwe do wykrycia za pomocą lupy);

b) tłoczenie – wyczuwalna dotykiem wypukłość lub wklęsłość powierzchni elementów stałych, stanowiąca zabezpieczenie lub będąca elementem wspomagającym rozpoznanie dla osób niewidomych lub słabo widzących;

c) grawerowanie wypukłe – wypukłości wyczuwalne dotykiem na powierzchni karty elementów personalizacyjnych (m.in. wspomaga rozpoznanie dla osób niewidomych lub słabo widzących);

d) mikrodruk + błąd w mikrodruku - linie, motywy lub części tła składające się z niewidocznych gołym okiem drobnych znaków (0,3 mm) w postaci np. liter, a możliwych do rozpoznania przy pomocy lupy.

Jako wartym rozważenia, w kontekście kart KUZ, KSA i KSM, jest wymaganie 8.2.2.8.4 wspomnianej umowy europejskiej (dotyczącej międzynarodowego przewozu drogowego towarów niebezpiecznych), które stanowi, że zaświadczenie powinno być w języku urzędowym, a w sytuacji gdy nie jest to język angielski, francuski lub niemiecki, tytuł zaświadczenia i kilka innych opisów pól powinien być również w jednym z tych trzech języków. W naszym przypadku wybrano, że obok nazwy polskiej jest również angielska i sugerujemy przyjęcie analogicznego podejścia w przypadku kart KUZ, KSA i KSM. Wydaje się, że ten aspekt nie musi być regulowany na poziomie ustawy, a jednie w rozporządzeniu wykonawczym, precyzującym m.in. warstwę graficzną karty.

Problem przygotowania wzoru graficznego

Projekt nowelizacji ustawy o sioz i innych ustaw wprowadza obowiązek określenia wzoru graficznego karty KUZ (i KSA) w rozporządzeniu wykonawczym. W związku z tym wydaje się niezbędne, aby była to firma wyłoniona w oddzielnym postępowaniu. Koszt wykonania projektu wzoru graficznego to kwota rzędu max. kilkunastu tys. zł, czyli można to zlecić bez konieczności uruchamiania skomplikowanej procedury przetargowej. Zwracamy jednocześnie uwagę, że osoba przygotowująca wzór graficzny powinna mieć wiedzę o ograniczeniach, jakie wprowadzają urządzenia personalizujące. Stąd zasadnym jest powierzenie zlecenia podmiotowi, który dysponuje doświadczeniem w personalizacji kart, w tym dysponuje certyfikatem INTERGRAF dot. „bezpiecznych dokumentów", ale jednocześnie musi być zagwarantowane, iż wykonanie tego zlecenia nie wykluczy go z przyszłego postępowania na dostawę spersonalizowanych kart KUZ/KSA.

DODATEK A – ANALIZA MECHANIZMÓW KRYPTOGRAFICZNYCH

A.1. Analiza mechanizmów kryptograficznych

Analiza dotyczy matematycznych właściwości algorytmów, pominięte są kwestie odporności implementacji na ataki typu side-channel, ataki typu reverse-engineering itp. Złożoności ataków dotyczą standardowego modelu obliczeń. Przez „efektywne bezpieczeństwo" rozumiany jest nakład pracy odpowiadający systematycznemu przeszukaniu przestrzeni kluczy o podanej długości w bitach.

A.1.1. RSA

Przedmiotem rozważań jest schemat podpisu elektronicznego z załącznikiem lub podpisu cyfrowego opartego o RSA-2048. Realny scenariusz ataku to próba odtworzenia klucza prywatnego lub znalezienie sposobu na konstruowanie poprawnych podpisów dla nieznanego wyzwania na podstawie wcześniej poznanych podpisów innych wyzwań, podsłuchanych lub wybranych przez przeciwnika (czyli *chosen message attacks*, choć scenariusz wyboru wyzwań wydaje się być mało prawdopodobny). Nie jest zagrożeniem atak znajdujący kolizje dla użytej funkcji skrótu.

Z punktu widzenia teoretycznego poruszamy się w obszarze tzw. *generic chosen message attacks* lub *adaptive chosen message attacks* (zależnie od tego czy założymy, że przeciwnik zna klucz publiczny służący do weryfikacji czy też nie – oczywiście bezpieczniej przyjąć, że zna) gdzie celem przeciwnika jest *Universal forgery* albo *total break* (ponieważ w protokole uwierzytelnienia to nie przeciwnik wybiera wiadomość do sfałszowania).

Brak jest dowodów równoważności (wielomianowej redukcji) tego schematu z którymkolwiek ze znanych trudnych problemów obliczeniowych. W związku z tym, mimo oparcia schematu o problem RSA, możliwe są ataki łamiące ten schemat poprzez rozkład modułu na czynniki, w tym rozkład wykorzystujący dodatkowe informacje ujawnione poprzez postać wykładnika publicznego, pierwiastkowanie oraz ataki wykorzystujące sposób formatowania bloku danych przed podpisem. Poniżej omówione są w skrócie znane ataki.

A.1.1.1. Ataki ogólne

A.1.1.1.1. Faktoryzacja modułu

Najlepszy znany ogólny algorytm faktoryzacji (tj. niewykorzystujący wiedzy o szczególnej postaci rozkładanych liczb) to GNFS o złożoności podwykładniczej. Prognozy dotyczące postępu w długości liczb rozkładanych tym algorytmem mogą być dokonywane na podstawie publikowanych wyników jego działania (czasu działania i użytych mocy obliczeniowych). Można też wyciągać wnioski z wyników działania odmiany tego algorytmu znanej jako SNFS (stosowanej do liczb postaci 20r-e). Ostatnio opublikowane wyniki omówione są poniżej.

Istnieją też niezweryfikowane w praktyce próby szacowania nakładów na faktoryzację w oparciu o nowatorskie rozwiązania sprzętowe. Ważnymi przykładami z tej kategorii prac są m.in. prace

- *Factoring Large Numbers with the TWIRL Device Adi Shamir and Eran Tromer z Crypto 2003* (dotyczy fazy odsiewania),
- *Arjen K. Lenstra, Adi Shamir, Jim Tomlinson, Eran Tromer, Analysis of Bernstein's factorization circuit, z Asiacrypt 2002* (dotyczy fazy znajdowania zależności).

W pierwszej z tych prac zaprezentowano propozycję sprzętowego układu realizującego fazę odsiewania algorytmu NFS, dzięki któremu kosztem ok. 10M USD odsiewanie dla 1024-bitowego modułu RSA trwałoby (wg szacunków autorów) ok. 1 roku. W drugiej pracy przedstawiono argumenty przemawiające za tym, że wąskim gardłem w faktoryzacji przy użyciu NFS ze wspomaganiem sprzętowym jest właśnie faza odsiewania.

A.1.1.1.1.1. Rozkład liczby RSA-200 (663 bity) [RSA-200]

[email od jednego z uczestników projektu, T. Kleinjunga, z dnia 9 maja 2005]

Prace nad rozkładem trwały od końca roku 2003 do roku 2005. Faza odsiewania wymagała nakładów obliczeniowych wynoszących szacunkowo 55 lat na procesorze Opteron 2.2 GHz. Faza znajdowania zależności zajęła ok. 3 miesięcy przy użyciu klastra 80 procesorów Opteron 2.2 GHz.

A.1.1.1.1.2. Rozkład RSA-640 (640 bitów)

Prace nad rozkładem trwały ok. 5 miesięcy kalendarzowych. Nakłady obliczeniowe wyniosły ok. 30 lat procesora Opteron 2.2 GHz. Składały się na to m.in. faza odsiewania która zajęła 3 miesiące na 80 procesorach Opteron 2.2 GHz i faza znajdowania zależności, która zajęła ok. 1,5 miesiąca na klastrze 80 procesorów Opteron 2.2 GHz.

A.1.1.1.1.3. Rozkład RSA-768 (768 bitów)

Prace nad rozkładem trwały ok. 2,5 roku. Nakłady obliczeniowe wyniosły ok. 1500 lat pracy procesora Opteron 2.2 GHz 2 GB RAM. Składały się na to m.in. faza odsiewania, która zajęła 6 miesięcy na klastrze 80 procesów Opteron 2.2 GHz i faza znajdowania zależności, która zajęła ok. 24 miesiące i była wykonywana na setkach maszyn.

A.1.1.1.1.4. Rozkład liczby większej niż 1024 bity o szczególnej postaci przy użyciu SNFS [2007/205]

Eprint 2007/205 A kilobit special number field sieve factorization

Opisane są kroki prowadzące do rozkładu liczby Mersenne'a 2^{1039}-1 przy użyciu SNFS. Faza odsiewania trwała ok. 6 miesięcy i wymagała nakładów równoważnych ok. 100 lat pracy procesora Opteron 2.2 GHz. Faza znajdowania zależności została zrealizowana na 2 klastrach: jednym składającym się ze 110 dwurdzeniowych procesorów Pentium D 3GHz i drugim składającym się z 96 czterordzeniowych procesorów Dual Core2Duo 2.66 GHz. Nakłady obliczeniowe w tej fazie można przyrównać do ok. 35 lat na pojedynczym rdzeniu procesora Pentium D lub 56 lat na pojedynczym rdzeniu procesora Dual Core Duo. Obliczenia w tej fazie trwały 69 dni. Wg autorów nakład obliczeniowy wystarczyłby do sfaktoryzowania 700-bitowego modułu RSA. W ocenie autorów fakt, że postępy w faktoryzacji zachodzą systematycznie oznacza, że RSA-1024 będzie bezpieczne przez okres nie dłuższy niż kilka lat.

A.1.1.1.1.5. Dalszy rozwój algorytmu GNFS

Postęp w szybkości działania algorytmu GNFS oczekiwany jest głównie w wyniku zastosowania sprzętowego wspomagania niektórych faz tego algorytmu. Szereg propozycji

takiego wspomagania zostało opublikowanych, jednak jak na razie wiadomo o tylko jednej próbie praktycznego wykorzystania takich urządzeń (grupa z Fujitsu, 2006 r.). Przykłady propozycji znaleźć można m.in. w:

Factoring Large Numbers with the TWIRL Device, Adi Shamir and Eran Tromer, Crypto 2003
Eprint 2006/109 A Simpler Sieving Device: Combining ECM and TWIRL
Eprint 2006/403 Non-Wafer-Scale Sieving Hardware for the NFS: Another Attempt to Cope with 1024-bit

Zabezpieczeniem przed atakami opartymi na ogólnych metodach faktoryzacji jest użycie dostatecznie dużych liczb. W chwili obecnej wydaje się, że 1024 bitowe moduły RSA jeszcze przez co najmniej kilka lat będą znajdować się poza granicą możliwości rozłożenia ich na czynniki pierwsze przy zastosowaniu publicznie znanych metod i racjonalnych nakładów obliczeniowych. W kartach KUZ, KSA i KSM stosowany będzie algorytm RSA o długości kluczy 2048 bitów, co czyni powyższe ataki bezprzedmiotowymi.

A.1.1.1.2. Pierwiastkowanie

Nie są znane efektywne metody obliczania pierwiastków modulo duża liczba złożona bez znajomości jej rozkładu (patrz [ECRYPT], str. 12). Najlepszą znaną metodą jest rozłożenie modułu na czynniki i wyznaczenie wykładnika prywatnego. Szczególny przypadek zachodzi jedynie wtedy gdy liczba, z której ma być wyciągnięty pierwiastek e-tego stopnia, jest e-tą potęgą w liczbach całkowitych. W przypadku protokołu uwierzytelnienia przy losowo wybieranych wyzwaniach prawdopodobieństwo takiego przypadku jest jednak skrajnie małe i pomijalne.

Odmianą pierwiastkowania mającą zastosowanie do odtwarzania wiadomości jest metoda opracowana przez Coppersmitha (i równoważna jej metoda Howgrave-Grahama) – nie ma ona jednak zastosowania do problemu fałszowania podpisów, gdyż warunkiem jej użycia jest znajomość znacznej części bitów wiadomości lub ograniczenie wartości rozwiązań do małych liczb.

Zabezpieczeniem jest losowe generowanie wyzwań, zapewnienie odporności przed faktoryzacją oraz stosowanie schematu podpisu z formatowaniem.

A.1.1.1.3. Ataki wykorzystujące homomorficzne właściwości algorytmu RSA

W tej klasie ataków znajdują się ataki mające na celu utworzenie poprawnego podpisu RSA bez znajomości klucza prywatnego. Podpis tworzy się jako pewną kombinację wcześniej zdobytych par {wiadomość, podpis}. Pierwowzorami takich ataków są ataki opisane przez Davida oraz Odlyzko i Desmedta, a ich rozwinięciem metody zastosowane przez Corona, Naccache, Grieu i innych do ataku na schemat 9796-1. Zabezpieczeniem przed tego typu atakami jest odpowiednie przygotowanie (formatowanie) bloku danych przed podpisem. Jak dotąd nie pokazano tego typu ataków na format podpisu stosowany w normie PKCS#1. W pracy [CorGri] odniesiono się do takiej ewentualności, ale uwarunkowanej pewnymi wymaganiami co do postaci modułu, które czynią taką ewentualność mało prawdopodobną (skuteczniejsze w tym przypadku będą inne metody ataku).

Zabezpieczeniem jest schemat PKCS#1, wobec którego nie zostały przedstawione skuteczne ataki oraz dobór kluczy RSA zgodnie z powszechnie akceptowanymi zaleceniami.

A.1.1.1.4. Inne

Trywialną metodą sfałszowania podpisu bez znajomości klucza prywatnego dla tzw. „surowego RSA" jest wybranie losowej liczby s i podniesienie jej do potęgi równej

wykładnikowi publicznemu. Otrzymany wynik i liczba s tworzą poprawną parę {wiadomość, podpis}. Atakujący nie ma jednak kontroli nad wiadomością.

Zabezpieczeniem jest stosowanie formatowania zapewniającego, że poprawne wiadomości są dostatecznie rzadkie, aby uczynić taki atak nieefektywnym. Przykładem takiego formatowania jest schemat PKCS#1.

A.1.1.2. Ataki na szczególne przypadki

W tej części zestawiono wybrane prace przedstawiające metody ataku na algorytm RSA dla pewnych szczególnych przypadków (prac o podobnym charakterze jest więcej; te, które podano są w miarę reprezentatywne). Należy zaznaczyć, że w większości przypadków losowo (i zgodnie z zaleceniami) wybrane klucze mają znikome prawdopodobieństwo trafienia w zbiór kluczy stanowiący specyfikację każdego takiego przypadku.

A.1.1.2.1. Małe wykładniki publiczne

W przypadku użycia małego wykładnika publicznego i błędu w implementacji schematu PKCS#1 v1.5 polegającego na nie dość rygorystycznej weryfikacji składni podpisanego bloku po stronie odbiorcy, możliwe jest sfałszowanie podpisu metodą zaproponowaną przez D. Bleichenbachera. Opis podatności znaleźć można na liście Bugtraq pod numerem 19849. Ten atak nie dotyczy algorytmu RSA jako takiego, ale niepoprawnej implementacji schematu podpisu elektronicznego opartego na tym algorytmie. Jednak ze względu na konsekwencje został w tym miejscu przywołany.

Zabezpieczeniem przed takim atakiem jest poprawna implementacja schematu PKCS#1 (istotne jest, aby skrót wiadomości był usytuowany na najniższych (tj. najmniej znaczących) pozycjach podpisanego bloku. Jeżeli tak nie jest, to procedura weryfikacji podpisu powinna potraktować podpis jako nieważny.

A.1.1.2.2. Wykładniki publiczne szczególnej postaci

Eprint 2006/093 RSA and a higher degree Diophantine equation, Abderrahmane Nitaj
Eprint 2006/235 Application of ECM to a class of RSA keys, Abderrahmane Nitaj

W obu pracach przedstawiono pewne klasy wykładników publicznych scharakteryzowanych przez pewne równanie diofantyczne. Spełnienie tego równania przez wykładnik publiczny pozwala na szybką (o złożoności wielomianowej) faktoryzację modułu.

Zabezpieczeniem jest generowanie kluczy w oparciu o losowo wybrane liczby pierwsze. Prawdopodobieństwo wystąpienia takiego przypadku jest znikomo małe przy losowo generowanych kluczach.

A.1.1.2.3. Małe wykładniki prywatne

D. Boneh, G. Durfee, Cryptanalysis of RSA with private key d less than N0,292, IEEE Transactions on Information Theory 46 (2000), 1339-1349

Pokazane jest, że stosowanie krótkich wykładników prywatnych pozwala na ich łatwe odtworzenie ze znanych wartości modułu i wykładnika publicznego. Autorzy przewidują, że właściwym progiem dla tego rodzaju ataku jest $N^{0,5}$.

Zabezpieczeniem jest generowanie kluczy w oparciu o losowo wybrane liczby pierwsze. W takim przypadku prawdopodobieństwo uzyskania wykładnika prywatnego mniejszego niż pierwiastek kwadratowy z modułu jest pomijalnie małe.

A.1.1.2.4. Wykładniki prywatne szczególnej postaci

J. Blomer, A. May, A generalized Wiener attack on RSA, In Practice and Theory in Public Key Cryptography (PKC 2004), Lecture Notes in Computer Science, Springer-Verlag 2947 (2004) 1-13.

Autor pokazuje, że gdy wykładnik prywatny ma postać $-xy^{-1}$ przy odpowiednio małych x, y, wtedy istnieje algorytm rozkładający moduł RSA na czynniki w czasie wielomianowym.

Zabezpieczeniem jest generowanie kluczy w oparciu o losowo wybrane liczby pierwsze. W takim przypadku prawdopodobieństwo uzyskania wykładnika prywatnego o podanej postaci jest znikome.

A.1.1.2.5. Czynniki modułu szczególnej postaci

A.1.1.2.5.1. Mała różnica między czynnikami

Klasyczna metoda faktoryzacji Fermata (lub metoda Lehmana) może być skuteczna jeżeli różnica między czynnikami modułu jest mała. Pewnym uogólnieniem tego faktu jest twierdzenie Coppersmitha mówiące o tym, że mając aproksymację czynnika o dokładności $o(N^{1/4})$ jest możliwe efektywne wyznaczenie tego czynnika.

Zabezpieczeniem jest generowanie kluczy w oparciu o losowo wybrane liczby pierwsze. W takim przypadku prawdopodobieństwo uzyskania czynników o małej różnicy jest pomijalnie małe.

A.1.1.2.5.2. Gładkie p-1, p+1 itp.

J.M. Pollard. "Theorems of Factorization and Primality Testing", Proceedings of the Cambridge Philosophical Society 76 (1974), pp. 521–528

H.C. Williams, A p+1 method of factoring, Math. Comp., 39, 225-234 (1982)

Dla odpowiednio niskiego progu gładkości B moduły będące iloczynem liczb p i q takich, że p±1 lub q±1 są B-gładkie poddają się faktoryzacji przy użyciu algorytmów takich jak algorytm p-1 Pollarda lub p+1 Williamsa. Warunki rozszerza się często o warunki na gładkość liczb p'-1 i q'-1 gdzie p'=p-1, q'=q-1 oraz odpowiednio p'+1 i q'+1.

Zabezpieczeniem jest dobór kluczy w oparciu o tzw. silne liczby pierwsze lub użycie na tyle dużych i losowych czynników modułu, przy których prawdopodobieństwo gładkości liczb występujących w podanych warunkach było na tyle małe, aby bardziej opłacalne z punktu widzenia kryptoanalityka było stosowanie ogólnych metod rozkładu na czynniki.

A.1.1.2.5.3. Małe czynniki

W takim przypadku oprócz GNFS zastosowane mogą być inne algorytmy rozkładu liczb całkowitych na czynniki pierwsze, takie jak algorytm rho Pollarda (lub jego modyfikacja zaproponowana przez Brenta) czy algorytm faktoryzacji oparty na krzywych eliptycznych (ECM), których złożoność czasowa zależy od rozmiaru najmniejszego czynnika rozkładanej liczby.

Zabezpieczeniem jest kontrolowanie rozmiaru czynników w procedurze generowania kluczy.

A.1.1.2.5.4. Inne

Eprint 2002/109 A New Class of Unsafe Primes, Qi Cheng

W pracy przedstawiony jest algorytm faktoryzacji działający w czasie wielomianowym dla liczb, których czynnik p ma tę cechę, że 4p-1 ma postać db^2 gdzie d należy do zbioru {3; 11; 19; 43; 67; 163}, b zaś jest liczbą całkowitą.

Zabezpieczeniem jest generowanie kluczy w oparciu o losowo wybrane liczby pierwsze. W takim przypadku prawdopodobieństwo uzyskania czynników o podanej postaci jest pomijalnie małe.

A.1.1.2.6. Moduły szczególnej postaci

Moduły postaci $2^k \pm c$ gdzie c jest małe są podatne na działanie odmiany NFS znanej jako SNFS.

Eprint 2006/107 The number field sieve for integers of low weight, O. Schirokauer

Również moduły o niskiej wadze Hamminga mogą stanowić potencjalną słabość – czas działania algorytmu NFS może być znacząco krótszy niż w ogólnym przypadku, choć o podobnej złożoności asymptotycznej.

Zabezpieczeniem jest generowanie kluczy w oparciu o losowo wybrane liczby pierwsze. W takim przypadku prawdopodobieństwo uzyskania modułu o podanej postaci jest pomijalnie małe.

A.1.1.2.7. Inne

Istnieją ataki odtwarzające cały klucz prywatny (wykładnik prywatny lub rozkład modułu na czynniki – w tej chwili wiadomo, że znajomość którejkolwiek z tych informacji pozwala w sposób efektywny i deterministyczny wyznaczyć drugą) na podstawie znajomości części tego klucza (*partial key exposure*). Mają one zastosowanie m.in. w sytuacjach, gdy atakujący ma dostęp do tzw. side-channel ujawniającego część bitów prywatnego wykładnika.

Zabezpieczeniem jest wprowadzenie mechanizmów zabezpieczających przed atakami typu side-channel.

Eprint 2002/183 Simple Backdoors for RSA Key Generation Claude Crepeau and Alain Slakmon

Autorzy przedstawiają sposoby generowania kluczy RSA pozwalające zawrzeć w kluczu publicznym informacje ułatwiające odtworzenie klucza prywatnego podmiotowi, który zaprojektował metodę generowania kluczy. Nawet, jeżeli nie ma on dostępu do poszczególnych instancji tej procedury. Wynika z tego, iż nie należy ufać kluczom RSA generowanym przez produkty (np. karty elektroniczne) dostarczane przez niezaufanych dostawców.

Zabezpieczeniem jest stosowanie własnych metod generowania kluczy, użycie kart posiadających stosowne certyfikaty lub dokładna analiza kodu procedury generowania kluczy. W przypadku projektu RUM II karty KUZ, KSA i KSM będą posiadać certyfikat SSCD wydany przez „ciało wyznaczone" w rozumieniu unijnych regulacji dot. podpisu elektronicznego, stąd zagrożenie takim atakiem nie występuje.

A.1.1.3. Zalecenia organizacji normalizacyjnych

A.1.1.3.1. ETSI (07/2011)

W dokumencie TS 102 176-1 dopuszcza się wykorzystanie algorytmu RSA z formatowaniem PKCS#1 v1.5, ale z zastrzeżeniem, że nie powinien on być stosowany w nowych aplikacjach, gdyż planowane jest wycofanie tego algorytmu z listy dopuszczonych. Jest jednak również uwaga (uwaga nr 2 na stronie 23), iż do maja 2011 nie był znany żaden rzeczywisty atak na tę metodę formatowania. Jednocześnie uznano, że metoda formatowania PKCS#1 v.1.5 może być stosowana przez co najmniej 6 lat.

Użycie kluczy RSA o długości 1024 bity nie jest dopuszczone dla par kluczy, których planowany okres ważności przekracza 1 rok, jednak zauważono (uwaga 2 na str. 30), że aż do maja 2011 r. nie odnotowano żadnego rzeczywistego ataku na RSA z taką długością kluczy. Zawarto jednak zastrzeżenie, iż konieczne jest zachowanie gotowości do unieważnienia tych par kluczy lub podjęcia innych kroków, jeżeli bezpieczeństwo RSA-1024 ulegnie pogorszeniu, np. zostaną ogłoszone rozkłady liczb z konkursu RSA.

ETSI uznaje, że na okres 3 lat dopuszczalne jest używanie RSA z kluczami o długości 1536 bitów, a w dającej się przewidzieć przyszłości (6 lat to okres „rozsądny"; w odróżnieniu od 10 lat, który to horyzont czasu określa się mianem „spekulacji") sugeruje wykorzystanie RSA z kluczami o długości 2048 bitów.

A.1.1.3.2. NIST

A.1.1.3.2.1. SP 800-78-3 Cryptographic algorithm and key sizes for personal identity verification (12/2010)

RSA-1024 jest dopuszczone do celów uwierzytelnienia, podpisu cyfrowego/elektronicznego i zarządzania kluczami, ale z zastrzeżeniem, że musi zostać wycofane do końca 2013 roku. Nałożone są również ograniczenia na wykładnik publiczny: powinien być większy lub równy 65537 i nie większy niż 2^{256}-1.

A.1.1.3.2.2. SP 800-57 Recommendation for key management Part 1: General (07/2012)

Efektywne bezpieczeństwo RSA-1024 jest ocenione jako równe 80 bitów, podlega ono wycofywaniu z użycia i jest dopuszczone do końca 2013 r. Natomiast RSA 2048-bitowy (efektywnie 112 bitów) można stosować do 2030 r. Przy bezpieczeństwie potrzebnym po 2030 r. należy używać RSA min. 3072-bitowego. Nie przewiduje się stosowania innych długości kluczy dla RSA.

A.1.1.3.2.3. FIPS 186-4 Digital signature standard (07/2013)

Dopuszcza się użycie RSA wg normy ANSI X9.31 lub PKCS#1. Minimalna długość modułu to 1024 bity. Podane są również kryteria (i metoda), które mają być używane przy generowaniu par kluczy m.in. postulowane jest, aby wykładnik publiczny e był nieparzystą liczbą naturalną z zakresu $2^{16} < e < 2^{256}$. Inne warunki dotyczą wykładnika prywatnego, odstępu między czynnikami i właściwości czynników chroniących przed rozkładem przy użyciu algorytmów p-1 lub p+1.

Dopuszcza się tylko 3 możliwe długości kluczy RSA wymienione w SP 800-57 (1024, 2048 i 3072), przy czym użytkownicy rządowi inni niż CA (podmioty świadczące usługi certyfikacyjne) powinni używać tylko pierwszych dwóch długości, natomiast CA z kolei powinny używać kluczy o długości nie mniejszej niż użytkownicy końcowi.

Najnowsza wersja normy (186-4) wprowadza zmiany dotyczące generowania bitów losowych i liczb losowych, wymagając stosowania RBG i RNG tylko z urządzeń certyfikowanych na zgodność z FIPS-140. Ponadto określa zasady ponownego wykorzystania ziarna do generowania liczb pierwszych p, q oraz zasady wyznaczania długości *salt* dla schematu RSASSA-PSS z PKCS#1, wersja 2.1, gdyż dotąd dla kluczy o długości 1024 bity były one niezgodne.

A.1.1.3.2.4. NIST SP 800-131A (01/2011)

Przewiduje się stosowanie RSA 1024-bitowego tylko do końca 2013 r.[6] Po tym terminie dopuszczalny jest RSA 2048-bitowy (odpowiadający 112-bitowemu poziomowi bezpieczeństwa).

A.1.1.3.3. ECRYPT (09/2012)

W raporcie zalecane jest wykorzystanie kluczy RSA o długości od 1024 w górę (przy czym 1024 bitów uznaje się jako absolutne minimum i może być używane tylko dla zapewnienia zgodności w istniejących systemach), w nowych systemach zaleca się stosowanie modułów o długości minimum 2432 bity. I wykładniki publiczne większe lub równe 65537 (choć dopuszczone są i mniejsze w szczególnych przypadkach dla RSA_KEM).

Dla zabezpieczeń nowych systemów zalecane jest stosowanie kluczy o długości co najmniej 2432 bity, na okres 30 lat kluczy o długości 3248 bitów, a na dłuższe okresy 15424 bity.

Jest zalecane, aby tam gdzie to możliwe stosować RSA-OAEP lub lepiej RSA_KEM, zamiast RSA-PKCS#1.5, dla którego nie ma dowodu bezpieczeństwa i znane są podatności.

Do podpisów elektronicznych zalecane jest stosowanie RSA_PSS.

W przypadku używania RSA-PKCS#1 wymagane jest używanie dużych, losowych wykładników publicznych i **zakazane jest używanie wspólnych kluczy do szyfrowania i podpisu.**

A.1.1.3.4. CRYPTREC (2003)

Zaleca stosowanie RSA-PSS oraz RSA PKSC#1 v.1.5 do podpisów elektronicznych oraz RSA-OAEP i RSA PKCS#1 v.1.5 (ten ostatni schemat dopuszczony tylko „na pewien czas" do użycia z „ostrożnością") do zapewnienia poufności.

A.1.1.4. Podsumowanie

Odporność na ataki inne niż faktoryzacja można zapewnić stosując odpowiednie zalecenia dotyczące doboru kluczy, np. te znajdujące się w dokumentach NIST.

Efektywne bezpieczeństwo RSA-1024 można ocenić (w ślad za [ECRYPT]) na 70-80 bitów o ile wykluczy się (poprzez dobór kluczy) możliwości ataków innych niż rozkład modułu na czynniki. Wg niektórych publikacji rozkład modułów RSA o tej długości wydaje się być technicznie możliwy, acz bardzo drogi (rzędu kilkuset milionów dolarów choć niektóre szacunki mówią o 10 milionach dolarów). Organizacje normalizacyjne wycofują się ze stosowania RSA-1024 (NIST zezwala na użycie do końca 2013 roku, ECRYPT tylko w istniejących systemach). Wobec tego w systemie NFZ zaleca się stosowanie RSA o długości co najmniej 2048 bitów.

Wykorzystanie ocen i zaleceń formułowanych w odniesieniu do użycia RSA przy podpisie cyfrowym lub elektronicznym do oceny bezpieczeństwa tego samego algorytmu użytego do uwierzytelnienia podmiotów wydaje się uzasadnione m.in. tym, że najczęstszy sposób realizacji protokołu uwierzytelnienia wykorzystuje mechanizm podpisu cyfrowego w formie

[6] w stosunku do wcześniejszej wersji tego dokumentu dokonano przedłużenia możliwości stosowania 1024-bitowego klucza o 3 lata, czyli *deadline* zmieniono z 2010 r. na 2013 r. z dodatkowym zastrzeżeniem, iż jednocześnie stosujący godzi się z pewnym, nieznacznym poziomem ryzyka

czarnej skrzynki. Nie wyklucza to oczywiście możliwości istnienia słabości w samym protokole, niezależnych od algorytmu podpisu lub związanych ze sposobem użycia tego algorytmu w protokole (tzw. problem „kompozycji kryptograficznych[7]" algorytmów).

A.1.2. DSA

A.1.2.1. Zalecenia organizacji normalizacyjnych

Digital Signature Algorithm został opublikowany w normie FIPS 186-3 i jest dopuszczony do stosowania w agencjach federalnych. Ostatnio zwraca się uwagę na sposób generowania parametrów tego algorytmu, ponieważ pewne (stosowane dotąd) deterministyczne generatory liczb losowych są obecnie wycofywane z użycia. Dokładniej, zalecenie NIST SP 800-90 zostało w styczniu 2012 r. zastąpione zaleceniem NIST800-90A.

A.1.2.1.1. ETSI (07/2011)

W dokumencie TS 102 176-1 dopuszcza się wykorzystanie algorytmu DSA o następujących parametrach (zgodnie z ISO/IEC 14888-3):

α = 1024, β = 160;
α = 2048, β = 224;
α = 2048, β = 256;
α = 3072, β = 256.

Przy czym wskazuje się wartość β = 160 jako nie zalecaną do stosowania w nowych aplikacjach ze względu na wycofywanie z użycia funkcji skrótu SHA-1. Ponadto ze względu na to, że stosowanie β = 224 nie daje żadnych korzyści w stosunku do β = 256, zaleca się stosowanie β = 256.

Klucz prywatny DSA składa się z publicznych parametrów *p*, *q* i *g* oraz unikalnej i nieprzewidywalnej liczby całkowitej *x*, 0<*x*<*q*, a także unikalnej i nieprzewidywalnej liczby całkowitej *k* , 0<*k*<*q*, która musi być ponownie generowana dla każdego podpisu. Jeśli rozkład liczb *k* znacząco odbiega od równomiernego, wówczas możliwe są ataki. Bleichenbacher opublikował atak prawie wyczerpujący, którego skuteczność zależy od odchylenia rozkładu k od równomiernego i liczby podpisów wytworzonych za pomocą pojedynczego klucza prywatnego. Wartość *k* musi być chroniona tak samo jak klucz prywatny, gdyż nawet częściowa informacja o k może umożliwić przeprowadzenie ataki [Nguyen, Shparlinski]

A.1.2.1.2. NIST

A.1.2.1.2.1. NIST SP 800-57 Recommendation for key management Part 1: General rev.3 (07/2012)

NIST SP 800-57 rev. 3 zaleca wycofywanie stosowania zabezpieczeń o poziomie bezpieczeństwa 80-bitów do końca 2013 roku i dopuszcza poziom bezpieczeństwa 112-bitowy do końca 2030 roku, po czym zaleca migrację do poziomu 128-bitowego.

Wobec powyższego wycofywane jest używanie DSA o parametrach (1024, 160), a po zakończeniu 2013 roku te parametry są niedozwolone. Zalecane jest stosowanie DSA o

[7] ang *cryptographic suties*

parametrach co najmniej (2048, 224) do końca 2030 roku, a po tym terminie zalecane jest stosowanie DSA o parametrach co najmniej (3071, 256).

A.1.2.1.2.2. NIST SP 800-131A (01/2011)

NIST SP800-131A Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths (01/2011) ocenia, że poziom bezpieczeństwa podpisów DSA odpowiadający 80 bitom w szyfrowaniu symetrycznym, zapewniany jest przy następującym doborze parametrów DSA: ((|p| ≥ 1024) i (|q| ≥ 160)) i ((|p| < 2048) lub (|q| < 224)). Poziom ten jest dopuszczony do końca 2013 roku. Od 2014 roku, poziom odpowiadający szyfrowaniu 112-bitowym szyfrem blokowym, to (|p| ≥ 2048) i (|q| ≥ 224)).

Do weryfikacji podpisów, złożonych przed zmianą poziomu bezpieczeństwa, możliwe jest stosowanie zestawu zapewniającego 80-bitowe bezpieczeństwo.

A.1.2.1.3. ECRYPT (09/2012)

Ecrypt II Yearly Report on Algorithms and Keysizes (2010-2011) (06/2011) określa bezpieczeństwo DSA z parametrami p≈1024 i q≈160 jako krótkoterminowe, natomiast przy parametrach q ≥ 256 oraz p ≥ 3072 jako średnioterminowe. Różnica w stosunku do NIST wynika z innego podejścia do określenia równoważnego poziomu bezpieczeństwa dla szyfrów asymetrycznych (tj. poziomowi 80-bitów dla szyfrów symetrycznych odpowiada poziom parametrów DSA: p=1248 i q=160, a poziomowi 112-bitów dla szyfrów symetrycznych odpowiada poziom parametrów DSA: p=2432 i q=224).

A.1.2.1.4. CryptRec (03/2003)

Cryptrec Report 2002 (03/2003) jest corocznie aktualizowany. Raport w oryginalnej wersji z 2003 r do podpisu elektronicznego zaleca stosowanie m.in. DSA. Niestety, ze względu na to, że kolejne raporty są dostępne w języku japońskim, nie mamy informacji o aktualnych zaleceniach dot. parametrów.

A.1.2.1.5. ISO/IEC 14888-3:2006 Amd. 1-2010

W normie ISO/IEC 14888-3:2006 Amd. 1 dopuszcza się wykorzystanie algorytmu DSA o następujących parametrach:

α = 1024, β = 160;
α = 2048, β = 224;
α = 2048, β = 256;
α = 3072, β = 256.

A.1.2.2. Podsumowanie

DSA jest stosowanym powszechnie i zalecanym przez organizacje normalizacyjne algorytmem podpisu elektronicznego. Przy właściwym doborze parametrów jego bezpieczeństwo nie budzi zastrzeżeń.

A.1.3. ECDSA (ECDH i ECIES)

Wszystkie 3 algorytmy oparte są o krzywe eliptyczne. Bazowym ciałem jest GF(p) gdzie p jest liczbą pierwszą długości 521 bitów.

W przypadku algorytmu ECIES istnieje dowód, że jest on CCA2-IND bezpieczny w modelu *random oracle* przy założeniu trudności problemu Gap-DH, pod pewnymi warunkami co do wyboru parametrów systemu (innych niż krzywa). Praktyczne bezpieczeństwo wiąże się jednak z problemem logarytmu dyskretnego w grupie, w której zdefiniowano ten system.

W przypadku ECDSA istnieją dowody bezpieczeństwa (tj. odporności na tzw. *existential forgery* przy *adaptive chosen Messager attack*) w tzw. generic group model, przy założeniu, że funkcja skrótu jest odporna na kolizje.

W praktyce bezpieczeństwo wszystkich 3 algorytmów jest powiązane z problemem logarytmu dyskretnego na krzywej eliptycznej.

A.1.3.1. Ataki ogólne

Jedynym znanym atakiem ogólnym jest wyznaczanie logarytmu dyskretnego w grupie punktów na krzywej eliptycznej. Nie są znane żadne algorytmy o złożoności mniejszej niż wykładnicza dla ogólnego przypadku. W praktyce stosuje się algorytmy takie jak algorytm Rho Pollarda, Lambda Pollarda, algorytm Shanksa, których złożoność czasowa wynosi O($\sqrt{n}$ $\sqrt{n}$) gdzie n jest rzędem grupy.

Postęp w wyznaczaniu logarytmów dyskretnych na krzywych eliptycznych można śledzić m.in. dzięki konkursowi firmy Certicom. Aktualnie największą krzywą (nad ciałem prostym), dla której udało się wyznaczyć logarytm dyskretny punktu wskazanego w konkursie to ECCp-109 dla 109-bitowego p. Nakłady obliczeniowe to 549 dni na ok. 10000 komputerów PC.

W dokumencie SEC-1 podano oszacowanie dot. wymaganych nakładów obliczeniowych na wyznaczenie logarytmu dyskretnego na krzywej eliptycznej rozmiaru 521 bitów równe $1.3*10^{66}$ MIPS-lat i zestawiono to z oszacowaniem podanym przez Odlyzko (w 1995 roku) dotyczącym mocy obliczeniowej reprezentowanej przez 0.1% całkowitej światowej mocy obliczeniowej w roku 2014 pracującej przez 1 rok równej 10^{11} MIPS-lat.

Zabezpieczeniem jest odpowiedni dobór rozmiaru grupy, w której prowadzone są obliczenia. W chwili obecnej 256-bitowe krzywe wydają się być poza zasięgiem dostępnych mocy obliczeniowych.

A.1.3.2. Ataki na szczególne przypadki

A.1.3.2.1. Szczególne krzywe eliptyczne

A.1.3.2.1.1. Krzywe supersingularne

Atak ma zastosowanie w przypadkach, gdy grupa, w której prowadzone są obliczenia ma rząd p+1 gdzie p jest charakterystyką ciała. Dla tych przypadków wyznaczanie logarytmu dyskretnego (poprzez przeniesienie problemu do pewnego ciała skończonego) ma złożoność podwykładniczą. Zastosowanie mają algorytmy Menezesa, Okamoto i Vanstone'a lub Freya i Ruecka. Praca Eprint 2007/343 zawiera propozycje rozszerzenia stosowanego kryterium MOV na tzw. subfield adjusted MOV.

Zabezpieczeniem jest wykrywanie i odrzucanie takich krzywych na etapie doboru krzywych.

A.1.3.2.1.2. Krzywe anomalne

Atak ma zastosowanie w przypadkach, gdy grupa, w której prowadzone są obliczenia ma rząd p, gdzie p jest charakterystyką ciała. Dla tych przypadków zastosowanie mają algorytmy Semaeva, Smarta oraz Satoha i Araki.

Zabezpieczeniem jest wykrywanie i odrzucanie takich krzywych na etapie doboru krzywych.

A.1.3.2.2. Generator grupy niskiego rzędu lub gładkiego rzędu

Tak dobrany generator grupy umożliwia siłowe wyznaczenie logarytmu dyskretnego lub zastosowanie algorytmu Pohliga i Hellmana wyznaczającego logarytmy dyskretne modulo dzielniki rzędu P.

Zabezpieczeniem jest dobór grupy (a więc i generatora P), której rząd jest dużą liczbą pierwszą.

A.1.3.2.2.1. Krzywe o niskiej liczbie klasowej maksymalnego rzędu

Istnieje (hipotetycznie) możliwość wyznaczania logarytmów dyskretnych na krzywej eliptycznej poprzez przeniesienie problemu do krzywej eliptycznej nad ciałem liczbowym.

Zabezpieczeniem jest losowy dobór krzywych eliptycznych dostatecznie dużych rozmiarów lub sprawdzanie odpowiedniego warunku podczas generowania krzywej. Normy ISO/IEC oraz ETSI zawierają wymogi (norma ETSI również wskazówki dotyczące implementacji odpowiedniego testu) dotyczące liczby klasowej (ang. class number).

A.1.3.3. Ataki na poszczególne algorytmy i ich warianty

A.1.3.3.1. ECDSA

W przypadku ECDSA należy podkreślić, że ujawnienie choćby jednej wartości k pozwala odtworzyć klucz sesyjny.

Zabezpieczeniem jest używanie dobrego generatora liczb losowych oraz uniemożliwienie ujawnienia wartości kluczy sesyjnych (np. zerowanie tych kluczy sesyjnych bezpośrednio po wykorzystaniu).

Użycie niepoprawnych krzywych eliptycznych lub kluczy publicznych przez podpisującego może być wykorzystane przez niego do wyparcia się podpisu.

Zabezpieczeniem jest używanie krzywych eliptycznych z wiarygodnego źródła lub ich walidacja oraz walidacja kluczy publicznych podpisującego.

Jest możliwe wygenerowanie takiej pary kluczy ECDSA, która pozwoli przedstawić 2 różne wiadomości pasujące do jednego podpisu. Taka cecha mogłaby posłużyć do wyparcia się autorstwa jednej z tych wiadomości. Ujawnienie tych wiadomości ujawnia jednak również klucz prywatny. Jest to więc możliwe do wykonania tylko przez właściciela klucza i nie jest traktowane jako prawdziwy atak na ECDSA.

Zabezpieczenie nie jest konieczne.

W SEC-1 wspomniana jest możliwość ataku na ECDSA poprzez tzw. elliptic curve semi-logarithm problem (ECSLP) [Brown01, Brown05b]. Niewiele wiadomo o możliwościach rozwiązywania tego problemu, nie jest on jednak trudniejszy niż logarytm dyskretny na krzywych eliptycznych.

Zabezpieczenie nie jest znane, jednak nie są również znane efektywne algorytmy wyznaczania semi-algorytmów.

A.1.3.4. Zalecenia organizacji normalizacyjnych

A.1.3.4.1. ETSI (07/2011)

Dopuszcza algorytm ECDSA m.in. nad ciałem GF(p). Stawia się następujące wymagania:

- liczba klasowa maksymalnego rzędu w pierścieniu endomorfizmów krzywej E powinna być równa co najmniej MinClass = 200.
- wartość r0: = min (r: q dzieli p^r-1) powinna być większa niż $r0_{Min}=10^4$.
- h = n/q powinno być mniejsze lub równe 4

Ponadto ustala się rekomendowane minimalne wartości parametrów, które pozwalają zapewnić bezpieczeństwo podpisów przez okres 1, 3 i 6 lat (dla okresu 10 lat minimalne wymagania nie są określone precyzyjnie, gdyż tak długi okres czasu pozwala jedynie na „spekulacje"): minimalny rząd grupy, w której prowadzone są obliczenia (q) ma wynosić co najmniej 2^{160} w przypadku wymogu zapewnienia bezpieczeństwa podpisu rzędu 1 roku oraz co najmniej 2^{224} w przypadku wymogu zapewnienia bezpieczeństwa podpisu w perspektywie 3 lat. Gdy okres ważności podpisów jest dłuższy (6 lat i więcej) rekomenduje się wielkość min. 2^{256}.

Dokument zawiera również stwierdzenie, że wszystkie krzywe eliptyczne podane w załączniku do FIPS 186-3, jak również w RFC 5639 („Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation"), spełniają wszystkie wymogi określone przez ETSI.

A.1.3.4.2. NIST

A.1.3.4.2.1. FIPS 186-4 Digital signature standard (07/2013)

Dopuszcza używanie algorytmu ECDSA i określa rekomendowane krzywe eliptyczne. Stwierdza, że użycie krzywej nad ciałem GF(p) dla |p|=521 bitów zapewnia poziom bezpieczeństwa równoważny algorytmowi AES-256.

A.1.3.4.2.2. SP800-56A rev. 2 (05/2013)

Dopuszcza użycie algorytmu ECDSA. Dopuszcza użycie krzywych eliptycznych o rozmiarach od 160 bitów, przy czym dla aplikacji rządowych należy stosować krzywe eliptyczne o rozmiarze min. 224 bity (o minimum tzw. „112-bitowym bezpieczeństwie" – patrz pkt 5.5.1.2 dokumentu NIST SP-800-56A rev. 2).

A.1.3.4.2.3. NIST SP 800-57-1 rev.3 (07/2012)

NIST SP 800-57 rev. 3 zaleca wycofywanie stosowania zabezpieczeń o poziomie bezpieczeństwa 80-bitów do końca 2013 roku i dopuszcza poziom bezpieczeństwa 112-bitowy do końca 2030 roku, po czym zaleca migrację do poziomu 128-bitowego.

Wobec powyższego wycofywane jest używanie ECDSA rozmiaru 160-223, a po zakończeniu 2013 roku te parametry są niedozwolone. Zalecane jest stosowanie ECDSA o rozmiarze co najmniej 224 do końca 2030 roku, a po tym terminie zalecane jest stosowanie ECDSA o rozmiarze co najmniej 256.

Siła algorytmów opartych na krzywych eliptycznych o rozmiarze większym niż 512 bitów została oceniona jako równoważna poziomowi bezpieczeństwa 256 bitów.

A.1.3.4.2.4. NIST SP 800-78-3 (12/2010)

Dla potrzeb uwierzytelnienia dopuszcza użycie krzywych eliptycznych o rozmiarach 256 lub 384 bitów. Dopuszcza użycie algorytmów ECDSA oraz ECDH (na potrzeby zarządzania kluczami uwierzytelniającymi).

A.1.3.4.2.5. NIST SP 800-131A (01/2011)

Nie ma ograniczeń w stosowaniu algorytmów opartych na krzywych eliptycznych o rozmiarach większych lub równych 224 bitów (efektywnie 112-bitowy). Do końca 2013 r. dopuszcza się (z pewnym ryzykiem) implementacje używające kluczy krótszych (między 160 i 223 bity).

A.1.3.4.3. ECRYPT (09/2012)

Stwierdzone jest, że dla poziomu bezpieczeństwa 2^n operacji grupowych należy używać kluczy o długości 2n bitów, co przekłada się na wymóg, aby rząd grupy, w której prowadzone są obliczenia był rzędu 2^{2n}.

Dla istniejących systemów dopuszczalne jest używanie krzywych eliptycznych o rozmiarach przynajmniej 160 bitów, dla nowych systemów zalecane jest wykorzystywanie grupy rzędu przynajmniej 224 bity.

A.1.3.4.4. NSA (2005)

W 2005 roku NSA ogłosiło tzw. suite B cryptography – zestaw jawnych algorytmów dopuszczonych do ochrony informacji klasyfikowanych i nieklasyfikowanych. Są wśród nich algorytmy ECDH, ECDSA (zgodnie z dokumentami NIST FIPS 186-2 oraz SP 800-56A) przy czym wymagane jest użycie krzywych eliptycznych nad ciałem prostym GF(p), gdzie p ma 256 lub 384 bity.

A.1.3.4.5. Cryptrec (2003)

Dopuszcza stosowanie ECDSA do podpisów elektronicznych oraz ECDH do uzgadniania kluczy. Wskazuje na wątpliwości dotyczące ECIES przy zapewnieniu poufności – jako podatne na atak z wybranym tekstem.

A.1.3.4.6. SECG (01/2010)

SEC-2 v.2.0 dopuszcza algorytmy ECIES, ECDH i ECDSA. SEC-2 dopuszcza rozmiary krzywych od 192 bitów do 521 bitów w przypadku krzywych nad ciałami prostymi. Krzywe o rozmiarze 192 bitów są ocenione jako zapewniające bezpieczeństwo na poziomie 96 bitów.

A.1.3.5. Podsumowanie

Pod warunkiem wyeliminowania tych przypadków, które pozwalają na efektywne wyznaczanie logarytmów dyskretnych algorytm ECDSA wydaje się, że należy wycofywać się z krzywych o rozmiarze 160-bitów i migrować na krzywe o rozmiarze co najmniej 192-bitów. Wobec tego w systemach NFZ należy wykorzystywać krzywe o rozmiarze co najmniej 192-bity.

A.1.4. AES

AES to symetryczny szyfr blokowy o długości bloku 128 bitów i 3 różnych długościach kluczy. W zależności od długości kluczy zmienia się liczba rund algorytmu:

- dla kluczy 128-bitowych (ozn. AES-128) ma 10 rund

- dla kluczy 192-bitowych (ozn. AES-192) ma 12 rund
- dla kluczy 256-bitowych (ozn. AES-256) ma 14 rund

A.1.4.1. Znane ataki

Większość ataków dotyczy wariantów AES uzyskanych np. poprzez ograniczenie liczby rund. Poniżej przedstawione jest zestawienie znanych ataków.

Algorytm	Liczba rund	Rodzaj ataku	Złożoność obliczeniowa[8]	Zajętość pamięci	Uwagi
AES-128	6	sumy częściowe [FKS00]	2^{44}	$6*2^{32}$CP	
	7	sumy częściowe [FKS00]	2^{120}	2^{128}-2^{119}CP	
	10	atak algebraiczny XSL [CP02]	2^{100}		
AES-192	7	RK DC z niemożliwymi różnicami [BDN06]	2^{94}	2^{56} CP	32 klucze
	7	RK DC z niemożliwymi różnicami [JD03]	2^{116}	2^{111} CP	2 klucze
	7	Square[FKL01]	2^{155}	$19*2^{32}$CP	
	7	DC z niemożliwymi różnicami [P04]	2^{186}	2^{92}CP	
	8	RK DC Rectangle [HKK05]	$2^{86.5}$	$2^{86.5}$ CP	4 klucze
	8	RK DC z niemożliwymi różnicami [BDN06]	2^{134}	2^{116} CP	32 klucze
	8	RK DC z niemożliwymi różnicami [BDN06]	2^{159}	2^{92} CP	32 klucze
	8	RK DC z niemożliwymi różnicami [JD03]	2^{183}	2^{88} CP	2 klucze
	8	RK DC z niemożliwymi różnicami [BDN06]	2^{184}	$2^{68.5}$ CP	32 klucze
	8	Square [FKL01]	2^{188}	2^{128}-2^{119} CP	
	9	RK DC Rectangle[9]	2^{125}	2^{86} CP	256 kluczy
	9	RK DC Rectangle [KHP07]	2^{182}	2^{85} CP	64 klucze
	10	RK DC Rectangle [KHP07]	2^{182}	2^{125} CP	256
	10	RK DC Rectangle [KHP07]	2^{183}	2^{124} CP	64
	12	RK DC [BK09]	2^{176}	2^{123} CP	
AES-256	9	RK DC [KHP07]	2^{120}	2^{99} CP	4
	10	RK DC Rectangle [BDK05]	$2^{171.8}$	$2^{114,9}$CP	256
	10	RK DC [KHP07]	$2^{172.8}$	$2^{113,9}$CP	64
	9	RK DC [BDKKS09]	2^{39}	2^{39} CP	
	10	RK DC [BDKKS09]	2^{45}	2^{44} CC	Related subkey
	14	RK DC [BK09]	2^{119}	2^{119} CP	

[8] Określana w liczbie szyfrowań danym rodzajem szyfru
[9] pojawiły się wątpliwości dotyczące wskazanego wyniku [KHP07]

Tabela 1. Ataki na AES

Legenda:
RK – atak z powiązanymi kluczami (ang. *related key*)
CP – wybrany tekst (ang. *chosen plaintext*)
DC – analiza różnicowa (ang. *differential cryptanalysis*)
LC – analiza liniowa (ang. *linear cryptanalysis*)
AA – atak algebraiczny (ang. *algebraic attack*)
DFA – różnicowa analiza błędów (ang. *Differential fault analysis*)

A.1.4.2. Przeszukiwanie kluczy

Oprócz ataków wskazanych powyżej możliwe są również ataki polegające na przeszukiwaniu przestrzeni kluczy (ataki siłowe), w tym w wariantach typu Time-Memory-Data Tradeoff. Poniżej przedstawiono znane ataki siłowe w zastosowaniu do AES:

Algorytm	Atak	Typ danych	Dane	Złożoność obliczeniowa[8]	Zajętość pamięci	Obliczenia wstępne[8]
AES-128	TMD [BS00]	ustalone znane teksty	2^{8}	2^{128}	2^{60}	2^{120}
			2^{20}	2^{100}	2^{58}	2^{108}
			2^{32}	2^{80}	2^{56}	2^{96}
			2^{43}	2^{84}	2^{43}	2^{85}
	Biham [B02]	ustalone znane teksty	2^{64}	2^{64}	2^{64}	2^{64}
AES-192	TMD [BS00]	ustalone znane teksty	2^{48}	2^{96}	2^{96}	2^{144}
			2^{64}	2^{128}	2^{64}	2^{128}
	Biham [B02]	ustalone znane teksty	2^{96}	2^{96}	2^{96}	2^{96}
AES-256	TMD [BS00]	ustalone znane teksty	2^{64}	2^{128}	2^{128}	2^{192}
			2^{85}	2^{170}	2^{85}	2^{170}
	Biham [B02]	ustalone znane teksty	2^{128}	2^{128}	2^{128}	2^{128}

Tabela 2. Ataki TMD na AES

A.1.4.3. Zalecenia organizacji normalizacyjnych

A.1.4.3.1. NIST

AES jest szyfrem, który zalecany jest do stosowania w agencjach rządowych do zapewnienie długookresowego bezpieczeństwa, tj. ponad 2030 r.

A.1.4.3.1.1. NIST SP 800-57 (07/2012)

Zalecenie NIST SP 800-57 dopuszcza wykorzystanie algorytmu AES-128, AES-192 i AES-256 obecnie i w okresie powyżej 2031 roku.

A.1.4.3.1.2. NIST SP 800-78-3(12/2010)

Zalecenie NIST SP 800-78-3 dopuszcza wykorzystanie algorytmu AES-128, AES-192 i AES-256 jako klucza uwierzytelniającego karty obecnie i po 2013 roku.

A.1.4.3.1.3. NIST SP800-131A (01/2011)

Zalecenie NIST SP 800-131A dopuszcza wykorzystanie algorytmu AES-128, AES-192 i AES-256 bez ograniczeń czasowych.

A.1.4.3.2. NSA (2005)

NSA Suite B w zestawie dopuszczonych algorytmów wskazuje AES. AES z kluczami 128-bitowymi ma być stosowany do ochrony informacji tajnych, a z kluczami 256-bitowymi do ochrony informacji ściśle tajnych.

A.1.4.3.3. Ecrypt (09/2012)

Ecrypt II Yearly Report on Algorithms and Keysizes (2011-2012) (09/2012) zaleca stosowanie AES do zapewnienia długookresowego poziomu bezpieczeństwa

A.1.4.3.4. Cryptrec (2003)

W raporcie Cryptrec dopuszczono stosowanie algorytmu AES w instytucjach rządowych.

A.1.4.4. Podsumowanie

Z dokonanego przeglądu ataków wynika, iż obecnie brak jest ataków mogących realnie zagrozić algorytmowi AES-256. Nawet użycie AES-128 dla celów takich jak określono w systemach NFZ wydaje się wystarczające.

A.1.5. 3DES

Algorytm DES, który stanowi podstawę 3DES, jako najbardziej popularny szyfr blokowy doczekał się wielu publikacji z zakresu kryptoanalizy. Począwszy od ataku Daviesa, poprzez analizę liniową i jej warianty, analizę różnicową i jej warianty, ataki algebraiczne i wiele innych. Jednak najskuteczniejszym atakiem na DES pozostał atak wyczerpujący, polegający na przeszukaniu całej przestrzeni kluczy. Wielokrotne szyfrowanie zastosowane w 3DES (wydłużyło szyfr z 16 do 48 rund) miało na celu zapobieganie atakom statystycznym, takim jak analiza liniowa, czy różnicowa, gdzie złożoność ataku wzrasta wraz z liczbą atakowanych rund. Cel ten został osiągnięty.

Wcześniej ukazała się publikacja [CW92] wykazująca, że DES nie tworzy grupy, wobec czego wielokrotne szyfrowanie nie może być zastąpione pojedynczym przekształceniem. Jednak ataki ze spotkaniem w środku [OW91, L98] wykazały, że poziom bezpieczeństwa wielokrotnego szyfrowania jest niższy niż wynikałoby to z długości wykorzystywanych w przekształceniu kluczy.

Bezpieczeństwo wielokrotnego składania szyfrowania oceniono w artykule [MH81].

Atak	Liczba rund	Złożoność obliczeniowa[8]	Zajętość pamięci	Uwagi
Atak ze spotkaniem w środku MITM [MO96]	3x16	2^{112}		$2^{k(m+1)/2}$ dla nieparzystej liczby szyfrowań m i klucza podstawowego szyfru o długości k
Atak na kolizje kluczy [B96]	3x16	2^{84}	2^{28}	$2^{mk/2}$ złożoność obliczeniowa ataku przy $2^{k/2}$ dostępnych parach (P,C)
Warianty ataku ze spotkaniem w środku [L98]	3x16	$1,3*2^{104}$		2^{32} par (P,C)
Atak ze spotkaniem w środku z powiązanymi kluczami [SW96]	3x16	2^{56}- 2^{72}		Przy jednej parze powiązanych kluczy

Tabela 3. Ataki na 3DES

A.1.5.1. Ataki z kompromisem TMD na 3DES

Rodzaj ataku	Dane	Złożoność obliczeniowa[8]	Zajętość pamięci	Obliczenia wstępne[8]
MTM bez obliczeń wstępnych [MO96]	1	2^{112}	2^{56}	0
GD-MTM [MO96]	1	2^{120}	2^{48}	0
GDD-MTM [CK09]	2^{14}	2^{104}	2^{64}	0
MTM z obliczeniami wstępnymi [MO96]	1	2^{56}	2^{113}	2^{113}
TMTO-MTM [CK09]	1	2^{80}	2^{101}	2^{113}
Rainbow-MTM [CK09]	1	2^{79}	2^{101}	2^{113}
BS [BM05]	2^{42}	2^{84}	2^{84}	2^{126}
BS-MTM [CK09]	2^{14}	2^{84}	2^{85}	2^{99}
TMD-MTM [CK09]	2^{14}	2^{71}	2^{99}	2^{99}

Tabela 4. Ataki na typu kompromis Time-Memory-Data na 3DES

A.1.5.2. Zalecenia organizacji normalizacyjnych

A.1.5.2.1. NIST

A.1.5.2.2. NIST SP 800-67 Rev. 1 Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher (01/2012)

NIST rekomenduje używanie TDES (z trzema różnymi, niezależnymi kluczami) do ochrony danych wrażliwych nieklasyfikowanych w agencjach rządowych. Do roku 2030 TDES, będzie funkcjonował równolegle z FIPS 197 (AES) jako zatwierdzony standard szyfrowania danych.

A.1.5.2.3. NIST SP 800-57 Rev. 3 Recommendation for Key Management — Part 1: General (07/2012)

W tym zaleceniu również nacisk jest położony na stosowanie TDES z trzema różnymi kluczami. Algorytm w tym wariancie dopuszczony jest do użytku do 2030 roku.

A.1.5.2.4. SP800-131A (01/2011)

NIST zaleca stosowanie poziomu bezpieczeństwa odpowiadającego co najmniej bezpieczeństwu szyfru symetrycznego o kluczu długości 80 bitów do końca 2013 roku. Jednym z dopuszczonych do użycia szyfrów ocenianych przez NIST, jako zapewniających poziom bezpieczeństwa 80-bitowy, jest 3DES z dwoma kluczami. Należy zwróci uwagę na fakt, że stosowanie tego szyfru jest ograniczone – jednym kluczem (2x56 bitowym) nie można szyfrować więcej niż 2^{20} bitów danych.

Od 2014 roku do 2030 r. zaleca się stosowanie poziomu odpowiadającego co najmniej 112 bitom – NIST ocenia, że taki właśnie poziom bezpieczeństwa jest realizowany przez 3DES z trzema niezależnymi kluczami. W przypadku użycia 3 różnych kluczy, jednym zestawem nie można szyfrować więcej niż 2^{32} 64-bitowych bloków (SP 800-67 Rev.1. Zał. E).

A.1.5.2.5. NIST SP 800-78-3(12/2010)

Analogiczne zalecenia można znaleźć w NIST SP 800-78-3 (10/2010).

A.1.5.2.6. ISO

TDES jest dopuszczony do użycia w normie ISO/IEC 18033-3:2005 *Information technology — Security techniques — Encryption algorithms — Part 3: Block ciphers.*

A.1.5.2.7. CryptRec

Cryptrec Report 2002 (03/2003) dopuszcza stosowanie 3DES z trzema różnymi kluczami, zgodnie z zaleceniami NIST SP800-67.

A.1.5.2.8. ECRYPT (09/2012)

Ecrypt II Yearly Report on Algorithms and Keysizes (2011-2012) dopuszcza stosowanie 3DES z trzema niezależnymi kluczami jako odpowiednie do zapewnienia bezpieczeństwa średnioterminowego i długoterminowego tj. 10-20 lat. Przy czym określa, że zalecenie to jest konserwatywne.

A.1.5.3. Podsumowanie

Obecnie wydaje się, że nie ma zagrożeń dla stosowania szyfru 3DES z trzema kluczami. Należy jednak przy tym stosować się do zalecenia NIST, aby nie szyfrować więcej niż 2^{32} 64-bitowych bloków jednym kluczem.

A.1.6. Ataki ogólne na iteracyjne funkcje skrótu

Do funkcji RIPEMD oraz funkcji z rodziny SHA mają zastosowanie ataki ogólne na strukturę Merkle-Damgarda. W strukturze Merkle –Damgarda, znalezienie jednej kolizji powoduje łatwe wyznaczenie następnych kolizji poprzez wydłużanie wiadomości. Atak na wyznaczenie drugiego przeciwobrazu (dla długich wiadomości) jest dla tej struktury zawsze szybszy niż atak wyczerpujący. Kolizje wielokrotne (tzn. wiele wiadomości dających taki sam skrót) można wyznaczyć niewiele większym kosztem niż kolizje pojedyncze. Ataki stadne (ang. *herding attacks*) można przeprowadzić znacząco mniejszym kosztem niż oczekuje się to w przypadku losowej funkcji.

W iteracyjnych funkcjach skrótu (takich jak struktura Merkle-Damgarda), możliwe jest znalezienie 2^t wielokrotnych kolizji ze złożonością $t*2^{n/2}$. W [J04] zaproponowano składanie t kolizji znalezionych w wyniku ataku z paradoksem dnia urodzin. Każda z kolizji pozwala wybrać wiadomość z pary wiadomości, a wybór możliwy jest t-razy. Dzięki temu można skonstruować zbiór 2^t różnych wiadomości składających się z t bloków i wszystkie te wiadomości będą dawały ten sam skrót.

Wyznaczeniu drugiego przeciwobrazu dla iteracyjnej funkcji skrótu, dla długich wiadomości opisano w [W84] – dla wiadomości o długości 2^k wymaga 2^{n-k} prób. Dalsze badania prowadzone w tym kierunku [D99] i [KS05] opisują sposób konstrukcji i wykorzystanie w ataku multikolizji, z kolizji pomiędzy blokami wiadomości o różnych długościach. Wyznaczenie drugiego przeciwobrazu dla wiadomości o długości 2^k, wymaga $2^{n-k+1}+k*2^{n/2+1}$ obliczeń. Rozwinięciem tej klasy ataków, w połączeniu z koncepcją [KK08], jest publikacja [AB08], gdzie wyznaczenie drugiego przeciwobrazu wymaga $5*2^{2n/3}+2^{n-k}$ obliczeń (nieznacznie więcej niż w poprzednim przypadku, ale atak jest znacznie bardziej elastyczny).

Atak wyznaczający przeciwobraz powinien mieć złożoność (w idealnym przypadku wynoszącą 2^n). Jednak dla funkcji opartych na strukturze iteracyjnej, atak stadny (ang. *herding attack*) [KK08] można przeprowadzić przy $2^{(2n-5)/3}$ obliczeniach.

A.1.7. RIPEMD-160

A.1.7.1. Ataki na RIPEMD-160

Przetwarzanie w funkcji kompresji wykorzystywanej w RIPEMD-128 i RIPEMD-160 odbywa się w dwóch równoległych gałęziach. Aktualizacja stanu wymaga obliczenia wartości dwóch gałęzi i połączenia ich wyjść. Ze względu na to, że stan wewnętrzny jest dwukrotnie dłuższy niż wyjście funkcji skrótu oraz na trudność jednoczesnego kontrolowania dwóch gałęzi, bardzo mało publikacji jest poświęconych tym funkcjom skrótu.

W [SW12] przedstawiono charakterystyki różnicowe 2-rzędu dla RIPEMD-128 i RIPEMD-160 o ograniczonej liczbie rund. Na ich postawie można przeprowadzić skuteczny atak 47 kroków z 64 dla RIPEMD-128 oraz na 40-kroków z 80 dla RIPEMD-160.

W [MP06] badano własności różnicowe funkcji RIPEMD-128 i 160. W [OS11] i [WS11] badano ataki na przeciwobraz, w pierwszej pracy wykazano możliwość odtworzenia przeciwobrazu dla 31 kroków RIPEMD-160, w drugiej dla wewnętrznych 35 kroków RIPEMD-160. W obu przypadkach złożoności były bliskie atakowi wyczerpującemu.

Teoretyczne wyniki na wyznaczenie drugiego przeciwobrazu dla bardzo długich wiadomości przedstawiono w [KS08]

Liczba kroków	Rodzaj ataku	Złożoność obliczeniowa[8]	Uwagi
30	Wyznaczenie drugiego przeciwobrazu [OS10]	2^{155}	2^{16} pamięci
31	Wyznaczenie przeciwobrazu [OS10]	2^{155}	2^{17} pamięci, ostatnie 31 kroków
38	Atak z 4-sumami [SW12]	2^{42}	
40	Atak z 4-sumami [SW12]	2^{36}	Zaczynając od 2 rundy
40	Atak z częściową sumą 2-wymiarową [SW12]	2^{42}	
42	Atak z częściową sumą 2-wymiarową [SW12]	2^{36}	Zaczynając od 2 rundy
43	Atak z częściową sumą 2-wymiarową [SW12]	2^{151}	
51	Atak z częściową sumą 2-wymiarową [SW12]	2^{158}	Zaczynając od 2 rundy

Tabela 5. Ataki na RIPEMD-160

A.1.7.2. Zalecenia organizacji normalizacyjnych

A.1.7.2.1. ECRYPT (09/2012)

ECRYPT II Yearly Report on Algorithms and Keysizes (2011-2012) (09/2012) dopuszcza stosowanie RIPEMD-160. W porównaniu z SHA-1, uznaje RIPEMD-160 za bezpieczniejszą funkcję skrótu.

A.1.7.2.2. Cryptrec (2003)

Cryptrec Report 2002 (03/2003) dopuszcza stosowanie funkcji RIPEMD-160.

A.1.7.2.3. ISO (2004)

ISO/IEC 10118-3:2004 zaleca stosowanie funkcji RIPEMD-160.

A.1.7.3. Podsumowanie

Przedstawione ataki nie stanowią realnego zagrożenia dla funkcji skrótu RIPEMD-160. Funkcja ta wydaje się obecnie bezpieczniejsza niż SHA-1 i często jest tak przedstawiana w

zaleceniach organizacji normalizacyjnych i w innych niezależnych ocenach. W systemach NFZ można dopuścić stosowanie funkcji RIPEMD-160.

A.1.8. SHA (SHA-1 i SHA-2)

Aplikacje podpisujące aktualnie wykorzystywane w obszarze podpisów elektronicznych, również przy „kwalifikowanych usługach", realizują funkcję skrótu SHA-1, jak również zwykle mogą współpracować (realizować *padding* i szyfrowanie ciągu bitów przy pomocy algorytmu RSA) z innymi funkcjami skrótu z rodziny SHA-2 (SHA-256 i SHA-512). SHA-1, mimo powszechnego przekonania o „złamaniu", tak naprawdę do tej pory (sierpień 2014 r.) funkcja nie została skompromitowana poprzez znalezienie kolizji. Najbardziej efektywny atak na SHA-1 (znalezienie losowych kolizji) jest możliwy przy redukcji funkcji skrótu z 80 do 73 rund [EGA10]. Ponadto należy zauważyć, że wspomniany atak dotyczy *teoretycznej* odporności na kolizje (ang. *collision resistance*), czyli znalezienie dwóch jakichkolwiek ciągów, których skrót byłby identyczny. Natomiast praktyczny atak musiałby po pierwsze: pozwalać na znalezienie kolizji dla istniejącego skrótu (ang. *2nd pre-image resistance*), a po drugie ten skrót powinien dotyczyć wiadomości, która miałaby praktyczne (semantyczne) znaczenie, a nie byłaby tylko losowym ciągiem bitów. Takich ataków aktualnie nie znaleziono, a – poza wspomnianym [EGA10] – artykuły [AS09] i [CR08] mówią jedynie o ataku na „jednokierunkowość" SHA-1 przy redukcji algorytmu do 45-48 rund. Stąd aktualnie należy mówić o *zmniejszeniu marginesu bezpieczeństwa*, a nie o „złamaniu" algorytmu SHA-1. Tym niemniej Ministerstwo Gospodarki kończy konsultacje (wrzesień 2014) w sprawie zmiany rozporządzenia w sprawie warunków technicznych i organizacyjnych dla kwalifikowanych podmiotów (delegacja z ustawy o podpisie elektronicznym), w której zaproponowano stopniowe wycofanie SHA-1 w aplikacjach podpisujących, a pozostawienie możliwości jej stosowania tylko w aplikacjach weryfikujących podpisy elektroniczne.

SHA-256/512 to dedykowane funkcje skrótu, należące do rodziny SHA-2. Tak jak inne funkcje z tej rodziny realizują one klasyczne podejście do projektowania funkcji skrótu, zarówno w warstwie funkcji kompresji, jak i w warstwie wiązania bloków. Cechą odróżniającą te funkcje od popularnych funkcji skrótu jest podwojony stan wewnętrzny (1024 bity dla SHA-512) oraz relatywnie duża długość skrótu wynosząca 512 bitów dla funkcji SHA-512. Do tej pory odnotowano niewiele prac podejmujących kwestię bezpieczeństwa tych funkcji skrótu.

A.1.8.1. Praktyczne bezpieczeństwo

Artykuł [SS08] prezentuje atak na ograniczoną do 22-kroków funkcję SHA-512, tj. wskazuje kolizję zachodzącą ze średnim prawdopodobieństwem 2^{-5} (w najgorszym przypadku 2^{-9}). Autorzy zaprezentowali charakterystykę różnicową, którą muszą spełniać wiadomości kolidujące ze sobą i podali przykład kolidujących wiadomości.

W artykule [IMP08] autorzy przedstawili atak na SHA-256 i SHA-512 ograniczony do 24 rund odpowiednio z wszystkich 64 (SHA-256) i 80 (SHA-512) o złożoności 2^{18} oraz $2^{28,5}$. Ponadto wykazali nielosowe zachowanie funkcji SHA-256 wskazując na istnienie prawie kolizji przy dowolnym wyborze wektora początkowego. Nie przeprowadzili badań dla funkcji skrótu SHA-512, ale oczekują uzyskania podobnych wyników.

A.1.8.2. Zalecenia organizacji normalizacyjnych

W związku z istotnym zmniejszeniem marginesu bezpieczeństwa oferowanego przez funkcję skrótu SHA-1 wszystkie organizacje standaryzacyjne rekomendują zastosowanie w nowych aplikacjach funkcji skrótu z rodziny SHA-2. Jednocześnie podkreślają, że aktualnie stosowane rozwiązania z funkcją SHA-1 nadal pozostają bezpieczne, np. NIST SP 800-131A (01/2011). Przedłużył stosowalność algorytmu SHA-1 do 2013 r. mimo wcześniejszych zapowiedzi o konieczności jego wycofania. Ponadto SP 800-131A uznaje aktualnie bezterminowo bezpieczeństwo weryfikacji podpisów elektronicznych złożonych przy pomocy kompozycji kryptograficznych opartych o funkcje skrótu SHA-1, jak również pozwala na bezterminowe stosowanie tego algorytmu we wszystkich aplikacjach nie związanych ze składaniem podpisów elektronicznych.

A.1.8.2.1. ETSI TS 102 176-1 (07/2011)

SHA-1 jest zaakceptowana do użycia przez ETSI do skracania wiadomości o maksymalnej długości 2^{64}-1 bitów. Ze względu na opublikowane ataki na SHA-1 zalecane jest w nowych aplikacjach zaimplementowanie co najmniej SHA-224 lub SHA-256, aby była możliwość migracji na nowe funkcje skrótu, gdyby SHA-1 lub RIPEMD-160 okazały się za słabe.

W uwadze stwierdzono, że obecnie wszystkie znane ataki na kolizje w SHA-1 wymagają kontrolowania pewnych fragmentów bloku danych, które są skracane i znajomości bitów danych poprzedzających kontrolowane bity. Jest to realistyczne w scenariuszach ataków na składanie podpisów pod dokumentami (w szczególności, jeśli jakiś aktywny program może być ukryty w dokumencie). Z drugiej strony w certyfikatach X.509 istnieje możliwość zabezpieczenia przed takimi atakami przez zawarcie w nich odpowiedniej ilości entropii (bitów nieznanych i nieprzewidywalnych dla atakującego), umieszczonej przed jakimikolwiek bitami nad którymi atakujący mógłby mieć kontrolę.

A.1.8.2.2. NIST

A.1.8.2.2.1. FIPS 180-4 (03/2012)

Zawiera specyfikację funkcji skrótu dopuszczonych do użycia w agencjach rządowych SHA-1, SHA-224, SHA-512/224, SHA-256, SHA-512/256, SHA-384 i SHA-512. W kwestii bezpieczeństwa algorytmów odwołuje się do NIST SP800 107.

A.1.8.2.2.2. NIST SP800 107 Rev.1 (08/2012)

SHA-1 nie jest odpowiednia do wykorzystania w podpisach elektronicznych dla których wymagany jest 112-bitowy poziom bezpieczeństwa, ponieważ nie zapewnia odporności na kolizje na poziomie 112-bitów. Może być stosowana w funkcji HMAC, gdyż odporność na znalezienie drugiego przeciwobrazu jest na poziomie wyższym niż 112 bitowy.

Ponadto do stosowania w agencjach rządowych zalecane są pozostałe funkcje wymienione w normie FIPS 180-4: SHA-224, SHA-512/224, SHA-256, SHA-512/256, SHA-384 i SHA-512. Dla tych funkcji poziom bezpieczeństwa określany jest jako wynikający z ataków ogólnych, takich jak przeszukanie całej przestrzeni przeciwobrazów (dla znalezienia drugiego przeciwobrazu), czy też ataku z paradoksem dnia urodzin dla wyznaczenia kolizji.

A.1.8.2.2.3. NIST SP800-57 Rev. 3(07/2012)

SHA-1 jest dopuszczone do stosowania w podpisach elektronicznych na poziomie bezpieczeństwa 80-bitów tylko warunkowo. Wykazano bowiem, że bezpieczeństwo tej funkcji w zastosowaniu do podpisów elektronicznych jest niższe niż 80-bitów. Na tym poziomie (wycofywanym do końca 2013 roku) funkcja pozostaje w użyciu w istniejących aplikacjach. Po 2013 roku ma być funkcja skrótu z rodziny SHA-2.

SHA-1 jest zaakceptowane do użycia w HMAC, KDF (ang. *Key Derivation Key*) i w generowaniu liczb losowych. W tych zastosowaniach poziom bezpieczeństwa SHA-1 uznaje się za równy 128 bitom.

A.1.8.2.3. Ecrypt (09/2012)

ECRYPT II Yearly Report on Algorithms and Keysizes (2011-2012) (09/2012) dopuszcza stosowanie SHA-1 w celu zapewnienia bezpieczeństwa krótkookresowego, do zapewnienia bezpieczeństwa długookresowego wymaga stosowania SHA-2.

A.1.8.2.4. Cryptrec (2003)

Dopuszcza stosowanie funkcji SHA-1, SHA-256, SHA-384, SHA-512 zgodnie z FIPS180-2.

A.1.8.2.5. NSA (2005)

NSA Suite B zaleca stosowanie SHA-256 do ochrony informacji tajnych, a SHA-384 do ochrony informacji ściśle tajnych.

A.1.8.2.6. ISO

ISO/IEC 10118-3 zawiera specyfikację SHA-1.

A.1.8.3. Podsumowanie

Obecnie bezpieczeństwo funkcji skrótu SHA-256 i SHA-512 wydaje się być poza zasięgiem kryptoanalityków. Jakkolwiek NIST ogłosił w 2007 r. konkurs na nową funkcję skrótu, umownie nazwaną SHA-3. W 2010 r. do finału dostało się 5 kandydatów: BLAKE, Grostl, JH, Keccak i Skein. Zwycięzcą konkursu została funkcja Keccak. Jej analiza była praktycznie poza zasięgiem kryptoanalityków ze względu na duży stan wewnętrzny. Po zakończeniu konkursu, NIST ogłosił, że nowa funkcja jest rezerwowa, w stosunku do istniejących, gdyż jest oparta na zupełnie innej konstrukcji. W związku z tym złamanie rodziny funkcji SHA z dużym prawdopodobieństwem nie spowoduje złamania nowej funkcji. Obecnie nowa funkcja została znormalizowana. Powstała norma FIPS 202 SHA-3 Permutation-Based Hash Standard.

W systemach NFZ zaleca się stosowanie funkcji SHA-2 co najmniej o długości bloku 256 bitów. Funkcję SHA-1 można stosować tylko do celów zapewnienia wstecznej kompatybilności i generalnie należy wycofywać się z jej stosowania. Dla większej pewności co do bezpieczeństwa systemu warto rozważyć implementację funkcji SHA-3.

A.1.9. Literatura

A.1.9.1. Literatura dot. RSA

[ECRYPT] European Network of Excellence in Cryptography, D.AZTEC.6, Hardness of the main computational problems used in cryptography, 14.03.2007

[CorGri] Cryptanalysis of ISO/IEC 9796-1, D. Coppersmith, J.S. Coron, F. Grieu, S. Halevi, C. Jutla, D. Naccache, and J.P. Stern, Journal of Cryptology

A.1.9.2. Literatura dot. AES

[B02] E.Biham, „How to decrypt or even substitute DES-encrypted messages in 2^{28} steps", IPL, 2002.
[B05] A.Biryukov, „The boomerang attack on 5 and 6-rounds AES", AES4, 2005.
[BDK05] E.Biham, O.Dunkelman, N.Keller „Related-Key Boomerang and Rectangle attacks", Eurocrypt 2005.
[BDK06] E.Biham, O.Dunkelman, N.Keller „Related-Key Impossible Differential attack on 8-round AES-192", CT-RSA 2006.
[BS00] A.Biryukov, A. Shamir, „Cryptoanalytic Time\Memory\Data TradeOffs for stream ciphers", Asiacrypt 2000.
[CP02] N.T.Courtois, J.Pieprzyk, Cryptanalysis of block ciphers with overdefined system of equations, Asiacrypt 2002.
[FKL01] N.Fergusson, J.Kelsey, S.Lucks, i inni, „Improved cryptanalysis of Rijndael", FSE 2001.
[FKS00] N.Fergusson, J.Kelsey, B.Schneier i inni, „Improved cryptanalysis of Rijndael", FSE 2000.
[GM00] H.Gilbert, M.Minier, „A collisio0n attaca on 7 rounds of Rijndael", AES 2000.
[HKK05] S.Hong, J.Kim, G.Kim, S.Lee, B.Preneel, „Related-key rectangle attacks on reduced versions of SHACAL-1 and AES-192", FSE 2005.
[JD03] G.Jakimowski, Y.Desmedt, „Related-key differential cryptanalysis od 192-bit key AES variants", SAC 2003.
[KHP07] J.Kim, S.Hong, B.Preneel, „Related-key rectangle attacks on reduced rounds AES-192 and AES-256", 2007
[P04] R.Phan, „Impossible differential cryptanalysis of 7-rounds Advanced Encryption Standard", IPL Vol. 81, No. 1, Elsevier 2004
[BDKKS09] A.Biryukov, O.Dunkelman, N.Keller, D.Khovratovich, A.Shamir, "Key Recovery Attacks of Practical Complexity on AES Variants With Up To 10 Rounds", 2009
[BK09] A.Biryukov, D.Khovratovich, "Related-key Cryptanalysis of the Full AES-192 and AES-256", 2009

A.1.9.3. Literatura dot. ECC

[Bro05a] D. Brown, Generic groups, collision resistance, and ECDSA. Designs, Codes and Cryptography, 35:119–152, 2005.
[Bro05b] D. Brown, On the provable security of ECDSA. In Blake et al. [BSS05], pp. 21–40.

A.1.9.4. Literatura do 3DES

[B96] E. Biham, How to Forge DES-Encrypted Messages in 2^{28} Steps, Pragocrypt 1996.
[BM05] A. Biryukov, S. Mukkhopadhyay, P. Sarkar, Improved Time-Memory Trade-Off with Multiple Data, SAC 2005.
[CK09] J. Choy, K. Khoo, Ch.-W. Loe, Applying Time-Memory-Data Trade-Off to Meet-in-the-Middle Attack, ePrint Archive 020/2009.
[CW92] K. W. Campbell. M. J. Wiener, DES is not a Group, Crypto 1992.
[L98] Stefan Lucks: Attacking Triple Encryption, FSE 1998.
[MH81] R. Merkle, M. Hellman, *On the Security of Multiple Encryption*, Communications of the ACM, Vol 24, No 7, pp 465–467, July 1981.
[MO96] A. Menezes, P. C. van Oorshot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996.
[OW90] P. van Oorschot, M. J. Wiener, *A known-plaintext attack on two-key triple encryption*, EUROCRYPT'90.
[SW96] J. Kelsey, B. Schneier, D. Wagner, Key-Schedule Cryptanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES, Crypto 1996.

A.1.9.5. Literatura do ataków ogólnych na iteracyjne funkcje skrótu

[AB08] E. Andreeva, Ch. Bouillaguet, P.-A. Fouque, J. J. Hoch, J. Kelsey, A. Shamir, S. Zimmer – Second Preimage Attacks on Dithered Hash Functions, In Proceedings of EUROCRYPT, LNCS 4965, pp. 270-288, Springer, 2008
[D99] R.D. Dean – Formal Aspects of Mobile Code Security, Phd Thesis, 1999
[J04] A. Joux, Multicollisions in Iterated Hash Functions. Application to Cascaded Constructions, CRYPTO 2004.

[KK06] J. Kelsey, T. Kohno. Herding Hash Functions and the Nostradamus Attack, Eurocrypt 2006.
[KS08] J. Kelsey, B. Schneier, Second Preimages on *n*-bit Hash Functions for Much Less than 2^n Work
[W84] R. Winternitz, A Secure One-Way Hash Function Built from DES, IEEE Symposium on Security and Privacy 1984
[LM93] X. Lai, J. L. Massey, Hash Function Based on Block Ciphers, EUROCRYPT 1993

A.1.9.6. Literatura dot. RIPEMD-160

[ISO04] ISO/IEC 10118-3:2004, Information technology – Security techniques – Hash-functions – Part 3: Dedicated hash-functions,' International Organization for Standardization, 2004.
[KS08] J. Kelsey, B. Schneier, Second Preimages on *n*-bit Hash Functions for Much Less than 2^n Work
[MP06] F. Mendel, N. Pramstaller, Ch. Rechberger, V. Rijmen, On the Collision Resistance of RIPEMD-160, ISC 2006.
[OS10] Ch. Ohtahara, Y Sasaki, T. Shimoyama, Preimage attacks on step-reduced RIPEMD-128 and RIPEMD-160, Inscrypt 2010.
[SA09] Y. Sasaki, K. Aoki, Meet-in-the-middle preimage attacks on double-branch hash functions: Application to RIPEMD and others, ACISP 2009
[SW12] Y. Sasaki, L. Wang, 2-Dimension Sums: Distinguishers Beyond Three Rounds of RIPEMD-128 and RIPEMD-160?, zgłoszone do FSE 2012.
[WL05] X. Wang, X. Lai, D. Feng, H. Chen, X. Yu, Cryptanalysis of the hash functions MD4 and RIPEMD, EUROCRYPT 2005.
[WS11] L. Wang, Y Sasaki, W. Komatsubara, K. Ohta, K. Sakiyama. (Second) preimage attacks on step-reduced RIPEMD/RIPEMD-128 with a new local-collision approach, CT-RSA 2011.

A.1.9.7. Literatura dot. SHA

[EAG10] E.A. Grechnikov. Collisions for 72-step and 73-step SHA-1: Improvements in the Method of Characteristics. Cryptology ePrint Archive, Report 2010/413, http://eprint.iacr.org/2010/413, 2010
[AS09] K. Aoki and Y. Sasaki. Meet-in-the-middle preimage attacks against reduced SHA-0 and SHA-1. Proceedings of Crypto 2009, LNCS 5677, pp. 70-89
[CR08] C. De Canni_ere and C. Rechberger, Preimages for Reduced SHA-0 and SHA-1, Proceedings of Crypto 2008, LNCS 5157, pp. 179-202, Srpinger-Verlag
[IMP08] S. Indesteege, F. Mendel, B. Preneel, and C. Rechberger, "Collisions and other Non-Random Properties for Step-Reduced SHA-256," COSIC internal report, 2008
[SS08] S.K. Sanadhya, P. Sarkar, „22-step for SHA-2", ar xiv.0803.1220v1, 2008
[SS08a] S.K. Sanadhya, P. Sarkar, „Non-linear reduced round attaca against SHA-2 hash family", ePrint 2008/174
[SS08b] S.K. Sanadhya, P. Sarkar, "Attacking Reduced Round SHA-256", ePrint 2008/142.

A.2. Oczekiwane ograniczenia czasowe bezpieczeństwa planowanych do implementacji algorytmów

Czas życia klucza kryptograficznego to okres czasu, w którym klucz jest autoryzowany do użycia przez uprawniony podmiot. Czas życia klucza kryptograficznego jest prawidłowo określony (zgodnie z NIST SP 800-57), gdy ogranicza się:

- ilość informacji zabezpieczonej tym kluczem, w celu ograniczenia potencjalnie dostępnego materiału do kryptoanalizy,
- ilość ujawnionych informacji, w przypadku kompromitacji tego klucza,
- użycie algorytmu kryptograficznego do szacowanego czasu życia tego algorytmu,

- czas na próby penetracji fizycznych, proceduralnych i logicznych mechanizmów kontroli dostępu zabezpieczających klucz przed nieautoryzowanym ujawnieniem,
- okres, w którym informacja może być skompromitowana przez nieumyślne ujawnienie materiału klucza osobom nieuprawnionym,
- czas dostępny dla obliczeniowo intensywnych ataków kryptoanalitycznych (w aplikacjach, w których długoterminowa ochrona klucza nie jest wymagana).

Wśród czynników wpływających na ryzyko ujawnienia kluczy należy wymienić:

- siłę mechanizmów kryptograficznych (algorytm, długość klucza, wielkość bloku danych, tryb pracy),
- otoczenie mechanizmów (np. w sprzętowej implementacji certyfikowanej na zgodność z FIPS 140 poziom 4, czy też w programowej implementacji),
- środowisko pracy (np. bezpieczne środowisko o kontrolowanym dostępie, biuro, czy też publicznie dostępny terminal),
- wolumen przesyłanych informacji lub liczbę transakcji,
- czas życia danych (jak długo istotne jest, aby dane były bezpieczne),
- funkcje bezpieczeństwa (szyfrowanie danych, podpis elektroniczny, tworzenie klucza lub odtwarzanie, ochrona klucza),
- metody wymiany kluczy (np. wprowadzanie ręcznie, ładowanie z urządzenia, w którym człowiek nie ma bezpośredniego dostępu do informacji, zdalna wymiana klucza poprzez PKI),
- proces aktualizacji kluczy lub uzyskiwania kluczy,
- liczba nodów sieci, które dzielą ten sam klucz,
- liczba kopii klucza i dystrybucja tych kopii,
- rotacja personelu (np. w szczególności personelu CA), i
- zagrożenie dla informacji (np. przed kim informacja jest chroniona, jakie są przewidywane techniczne możliwości i zasoby finansowe żeby przeprowadzić atak).

Inne aspekty wpływające na czas życia kluczy

Klucze wykorzystywane do informacji przesyłanej vs. przechowywanej:

- Jeśli klucze są wykorzystywane do zapewnienia poufności komunikacji, ich wymania następuje po krótszych okresach czasu niż w przypadku kluczy służących do przechowywania informacji. Jest to związane z kłopotliwą koniecznością przeszyfrowywania danych w przypadku zmiany kluczy służących po szyfrowania danych przechowywanych.

Koszt unieważnienia i zastąpienia kluczy

- W pewnych przypadkach koszt związany z wymianą klucza jest wysoki (np. konieczność deszyfrowania i ponownego szyfrowania nowym kluczem dużych baz danych lub rozproszonych baz danych, czy wymiana dużej liczby kluczy dla użytkowników rozproszonych geograficznie). W takich przypadkach wydatki na środki zabezpieczeń konieczne do zapewnienia dłuższych okresów ważności kluczy mogą być uzasadnione (tj. kosztowne i niewygodne fizyczne, proceduralne i logiczne środki kontroli dostępu, użycie odpowiednio silnej kryptografii, by zrekompensować dłuższe czasy życia kluczy, nawet jeśli skutkiem ich wprowadzenia będzie zwiększenie

obciążenia związanego z przetwarzaniem – dłuższy czas przetwarzania). W innych przypadkach okres życia klucza powinien być krótszy.

Czasy życia dla kluczy asymetrycznych

- Dla pary kluczy, każdy z kluczy ma swój własny okres ważności. Oznacza to, że każdy klucz może być użyty jako „nadawczy" aby zastosować zabezpieczenie kryptograficzne (np. wytworzyć podpis elektroniczny), albo „odbiorczy" aby przetwarzać zabezpieczoną informację (np. zweryfikować podpis elektroniczny), jednak nie obie te funkcje na raz. W przypadku, gdy klucze publiczne są dystrybuowane w certyfikatach, okres ważności obu kluczy nie musi być koniecznie jednakowy. Okres ważności klucza publicznego może być dłuższy niż klucza prywatnego, w szczególności dla zapewnienia długoterminowej weryfikacji podpisów elektronicznych.

Czasy życia dla kluczy symetrycznych

- Dla kluczy wykorzystywanych w kryptograficznych systemach symetrycznych, jeden klucz stosowany jest zarówno do zabezpieczenia (szyfrowania lub obliczenia kodu uwierzytelnienia wiadomości MAC), jak i do przetwarzania zabezpieczonej informacji (odszyfrowania lub weryfikacji kodu uwierzytelnienia wiadomości MAC). Okres ważności kluczy można podzielić na 2 okresy: okres *użycia nadawczego* i okres *użycia odbiorczego*. Łącznie dają one czas życia klucza.

A.2.1. Rekomendowane czasy życia kluczy

A.2.1.1. NIST

Rodzaj klucza	Okres życia klucza	
	Użycie nadawcze	Użycie odbiorcze
klucz prywatny podpisujący	1-3 lat	
klucz publiczny weryfikujący	Kilkanaście lat (w zależności od długości)	
Klucz prywatny do uwierzytelnienia	1-2 lata	
Klucz publiczny do uwierzytelnienia	1-2 lata	
Klucz prywatny do transportu kluczy	<= 2 lata	
Klucz publiczny do transportu kluczy	1-2 lat	

A.2.1.2. eCrypt (2012)

Rodzaj algorytmu	Okres życia klucza			
	2014-2020	2014-2030	2014-2040 rok	„przewidywalna przyszłość”[10]
symetryczny	96	112	128	256
asymetryczny	1776	2432	3248	15424
grupa algorytmiczna	1776	2432	3248	15424
krzywa eliptyczna	192	224	256	512
funkcja skrótu	192	224	256	512

A.2.1.3. ALGO (draft z lutego 2014)

Rodzaj algorytmu	Okres życia klucza			
	2015 rok	2017 rok	2020 rok	2030 rok
sha256-with-rsa	1536	2048	2048	nie wiadomo
RSASSA-PSS with mgf1SHA-1Identifier	1536	nie rekomendowane		
RSASSA-PSS with mgf1SHA-224Identifier	1536	2048	2048	dopuszczone
RSASSA-PSS with mgf1SHA-256Identifier	1536	2048	2048	dopuszczone
sha224-with-ecdsa	224	224	nie rekomendowane	
sha256-with-ecdsa	256	256	256	256
sha256-with-dsa	256	256	256	256

A.2.1.4. Rekomendacje BSI (2014)

Rodzaj algorytmu	Okres życia klucza		
	2013-2015	2016-2020	> 2020
asymetryczny	1976	1976	1976
grupa algorytmiczna	2048	2048	2048

[10] Z wyłączeniem kwantowego algorytmu Shora, umożliwiający rozkład na czynniki pierwsze liczby naturalnej N przy użyciu komputera kwantowego

krzywa eliptyczna	224	250	250
funkcja skrótu do weryfikacji	SHA-1 RIPEMD-160 >= SHA-224	>=SHA-256	>=SHA-256
funkcja skrótu do podpisu	>= SHA-224	>=SHA-256	>=SHA-256

A.2.2. Literatura

[1] NIST SP 800-57 Part 1, rev. 3 Elaine Barker, William Barker, William Burr,

[2] William Polk, and Miles Smid, Recommendation for Key Management – Part 1: General (Revision 3), July 2012

[3] ECRYPT II Yearly Report on Algorithms and Keysizes (2011-2012)

[4] Notification with regard to electronic signatures in accordance with the Electronic Signatures Act and the Electronic Signatures Ordinance (BSI, "Overview of Suitable Algorithms")

Koniec dokumentu