Protokoły Kryptograficzne

Notatki do testu

Arkadiusz Ostrzyżek

Contents

1 Definicja protokołu kryptograficznego

Protokół jest to szereg kroków, obejmujących dwie lub więcej stron, podejmowanych w celu realizacji zadania. Inaczej mówiąc jest to sekwencja działań z których każde musi być kolejno wykonane i żadne nie może być podjęte, zanim poprzednie nie zostanie ukończone. Protokół kryptograficzny jest to protokół wykorzystujący kryptografię.

2 Własności protokołu kryptograficznego

1. Każdy użytkownik musi go znać i kolejno wykonywać wszystkie kroki.
2. Każdy użytkownik musi zgodzić się na jego stosowanie.
3. Protokół nie może być mylący, każdy krok powinien być dobrze zdefiniowany i nie może wystąpić jakakolwiek szansa na nieporozumienie.
4. Protokół musi być kompletny, dla każdej możliwej sytuacji musi być podany odpowieni sposób postępowania.

3 Typy protokołów kryptograficznych

3.1 Protokoly arbitrażowe

Wymagają istnienia zaufanej trzeciej strony.

3.2 Protokoły rozjemcze

Protokół wymaga istnienia sędzi. Sędzia nie jest stałym uczestnikiem protokołu, jest on bezpośrednio zatrudniany tylko w przypadku sporów między stronami. Sędzia wydaje orzeczenia o poprawności albo nie dokonaniu transakcji.

Strony protokołu przyjmują bez zastrzeżeń wszystkie orzeczenia dotyczące: - prawdziwości wypowiedzi, - poprawności uczynków, - poprawności zakończenia przewidzianej dla danej strony części protokołu.

3.3 Protokoły samowymuszające

W tych protokołach strony porozumiewają się bezpośrednio. W przypadku oszustwa jednej ze stron, druga strona przerywa protokół. Nie jest wymagana zaufana trzecia strona.

4 Diffie-Hellman

4.1 Definicja

Diffy-Hellman (DH) służy do ustalenia klucza prywatnego używając jawnych kanałów komunikacji.

4.2 Wykonanie

0. Achilles i Bachus kanałem w sposób jawny ustalają p (moc zbioru) oraz g (generator).
1. Achilles i Bachus wybierają potajemnie liczby (s). Następnie wykonują peracje g^s % p = t. Wynik tych operacji przesyłają w sposób jawny sobie nawzajem.
2. Achilles i Bachus wykonują peracje na otrzymanych liczbach, t^s % p = f.

Wyniki tych operacji dadzą im ich nowy klucz do komunikacji. Będzie on taki sam, ponieważ $g^{s1}s2$ % p = $g^{s2}s1$ % p.

4.3 Atak MITM

Atakujący zna tylko: p, g, t1, t2. Oznacza to, że nie jest w stanie wykonać finalnej operacji, ponieważ t1^t2 % p != $g^{s1}s1$ % p. Atakujący musiałby w jakiś sposób pozyskać s jednej z osób, poprzez rozwiązanie logarytmu dyskretnego, który ma wysoką złożoność czasową.

5 Uwierzytelnianie

5.1 Definicja

nie podana w prezentacjach

Może być oparte o hasła, fizyczne klucze, karty, dane biometryczne, lokalizacje.

5.2 Przykłady

Opis protokołu: 1. użytkownik przesyła aktualny identyfikator xk 2. system sprawdza istnienie użytkownika o otrzymanym identyfikatorze, po czym żąda podania hasła czyli xk-1 3. użytkownik podaje hasło xk-1 4. system weryfikuje poprawność hasła, sprawdzając, czy f(xk-1) = xk, jeśli tak, to zapamiętuje xk-1 jako identyfikator przy następnym logowaniu.

6 Podział wiadomości

6.1 Definicja

Podział wiadomości (ang. Secret splitting) ma na celu takie ukrycie informacji pomiędzy n użytkownikami, aby odtworzenie danej wiadomości wymagało współpracy wszystkich uczestników (m.in. ich „części"). wiadomości).

6.2 Przykład

1. użytkownik T chce dokonać podziału wiadomości M między użytkowników A i B, w tym celu generuje ciąg losowy R o tej samej długości co M,

2. użytkownik T oblicza sumę modulo 2 ciągów M i R, tworząc P,
3. wysyła P użytkownikowi A , natomiast R użytkownikowi B (może również dokonać tego na odwrót),
4. użytkownicy A i B by odtworzyć wiadomość muszą wykonać sumę modulo 2 ciągu P i R.

7 Schematy progowe

7.1 Definicja

Innym zagadnieniem są podziały progowe, gdzie wiadomość dzielimy na „części" które rozdzielamy pomiędzy n użytkowników i ustalamy, że jeżeli zbierze się k lub więcej dowolnych użytkowników (k <= n), to mogą oni odtworzyć wiadomość. „Części" otrzymane z podziału wiadomości nazywa się cieniami.

7.2 Przykład

DO ZROBIENIA

8 Podpis niezaprzeczalny

8.1 Definicja

Składa się je pod dokumentem w podobnym celu jak zwykły podpis, jednak tym się różni od niego że sprawdzający poprawność podpisu musi skontaktować się z jego wytwórcą celem jego sprawdzenia.

Stawiający podpis ma kontrolę nad jego sprawdzaniem i sprawdzającymi.

8.2 Cechy

Użytkownik B nie może na podstawie otrzymanych danych z powyższych kroków przekonywać postronne osoby o poprawności podpisu Achillesa.

Każdy z pozostałych użytkowników chcących sprawdzić poprawność postawionego podpisu musi wykonać powyższe kroki protokołu osobiście, natomiast Achilles ma kontrolę nad tym, kto taką kontrolę podpisu chce zrealizować.

8.3 Przykład

Znana jest duża liczba pierwsza p i generator g. Achilles posiada klucz prywatny (e) i publiczny(d). Chce podpisać wiadomość m.

8.3.1 Podpisanie wiadomości:

Achilles generuje podpis: z = mˆe (mod p)

8.3.2 Weryfikacja podpisu

0. Bachus losowo wybiera a i b.
1. Bachus wybiera dwie liczby losowe a i b, obie mniejsze od p, przesyła do Achillesa wynik działania:

```
c = z^a*(g^x)^b (mod p)
```

2. Achilles oblicza x^-1 (mod p - 1) i przesyła do Bachusa wynik działania:

```
d = c^x^(-1) (mod p)
```

3. Bachus sprawdza, czy: `d = m^a*g^b (mod p).`

Poprawność działań widać po podstawieniu wszystkich działań:

```
(((m^x)^a)*((g^x)^b))^x^(-1) = (m^a)*(g^b)
```

9 Przesłanie niezaprzeczalne

9.1 Definicja

Realizowane jest gdy jedna ze stron ma do przekazania pewną wiadomość (ciąg bitów) za gratyfikacją, druga strona potrzebuje tej wiadomości, nie chce kupować „kota w worku" i strony nie ufają sobie.

9.2 Opis

1. Achilles wytwarza dwie pary kluczy publiczny/prywatny i przesyła oba klucze publiczne Bachusowi
2. Bachus wybiera klucz w algorytmie symetrycznym, losowo pobiera jeden z kluczy publicznych Achillesa i za jego pomocą szyfruje swój klucz algorytmu symetrycznego, przesyła zaszyfrowany klucz Achillesowi bez wskazania który z jej kluczy publicznych został wykorzystany do szyfrowania
3. Achilles deszyfruje klucz Bachusa, używając obu swoich kluczy prywatnych, w jednym z przypadków używa on poprawnego klucza i skutecznie deszyfruje klucz algorytmu symetrycznego Bachusa, ponieważ nie zna który klucz tego algorytmu jest poprawny (oba ciągi wyjściowe podobne są do ciągów losowych) oba ciągi są dla niej równoprawne
4. Achilles szyfruje, wykorzystując algorytm symetryczny, jedną wiadomość przy użyciu pierwszej wersji klucza i drugą za pomocą drugiej wersji klucza, oba wyniki przesyła do Bachusa;
5. Bachus deszyfruje obie wiadomości przy użyciu poprawnego klucza algorytmu symetrycznego, ale w wyniku otrzymuje tylko jedną z dwóch poprawną wiadomość;
6. po zakończeniu protokołu, gdy są znane oba możliwe wyniki przesłania Achilles może przesłać do B swój klucz prywatny, aby mógł on sprawdzić, czy on nie oszukuje.

10 Obliczenia z udziałem wielu stron

10.1 Definicja

Celem bezpiecznych obliczeń z udziałem wielu stron jest wyznaczenie wartości wybranej przez uczestników funkcji dla ich prywatnych wartości wejściowych

10.2 Własności

Pożądane własności: - Poufność; - Poprawność; - Niezależność od wartości wejściowych; - Gwarancja dostarczenia wyniku; - Uczciwość.

11 Obliczenia danych zaszyfrowanych

11.1 Definicja

Jest to grupa problemów polegających na tym, że zleca się osobie trzeciej obliczenia, a zarazem nie można ujawnić tej osobie argumentu obliczeń. Dla zlecających natomiast ważny jest wynik obliczeń na znanym tylko dla nich argumencie. Przykładem może być tu wyznaczanie logarytmu dyskretnego pewnej wartości x przez inne osoby, bez ujawnienia wartości x.

11.2 Przykład

Wyznaczenie logarytmu dyskretnego

1. Achilles wybiera liczbę losową r mniejszą niż p
2. Achilles wykonuje obliczenia `x' = x*g^r (mod p)`
3. Achilles prosi Bachusa o obliczenie wartości `e' = log(g, x') (mod p)`
4. Bachus oblicza e' i przesyła wynik do Achillesa
5. Achilles odtwarza e poprzez wyliczenie `e = e' - r (mod p - 1)`

12 Zobowiązania bitowe

12.1 Defiincja

Problem w tym przypadku polega na tym, że musimy ustalić pewne wartości, których początkowo nie można ujawnić, natomiast druga strona musi mieć pewność, że tych wartości nie zmienimy w trakcie realizacji zadania wspólnego. Na koniec wartości te są ujawniane celem konfrontacji.

12.2 Założenia

Założenia obiektów stanowiących zobowiązania bitowe: - Achilles może zobowiązać się co do postaci obiektów bitowych - Achilles może otworzyć dowolny obiekt bitowy, co do postaci którego zobowiązał się wcześniej, nie może on

jednak ujawnić obiektu bitowego, którego wartość dla Bachusa wynosiła by jednocześnie zero i jeden - Bachus nie może dowiedzieć się niczego o tym, w jaki sposób Achilles otwiera dowolny nie ujawniony obiekt bitowy, w stosunku do którego dokonał zobowiązania - obiekty bitowe nie zawierają żadnej innej informacji niż ta, która określa wartość zobowiązania bitowego, same obiekty bitowe, jak i proces zobowiązywania się i otwierania obiektu nie są skorelowane z jakąkolwiek inną informacją, którą Achilles pragnąłby utrzymać w tajemnicy przed użytkownikiem B.

12.3 Przykład

1. Achilles generuje dwa ciągi losowe R1 i R2,
2. Achilles tworzy wiadomość składającą się z jego ciągów losowych i bitu (bitów) b, który stanowi zobowiązanie,
3. Achilles oblicza wartość skrótu wiadomości, a Bachusowi wysyła wynik i jeden z ciągów losowych, H(R1, R2, b), R1. -> B Wartości te stanowią zobowiązanie strony A, Strona B nie może na podstawie skrótu i jednej wartości losowej

Dokończeniem protokołu powinny być następujące czynności: 1. Achilles przesyła do Bachusa pierwotną wiadomość: R1, R2, b 2. Bachus oblicza wartość skrótu wiadomości, porównuje ją z wcześniej uzyskaną, oraz porównuje R1 otrzymane wcześniej i obecnie, jeśli wartości te są zgodne, przyjmuje bity b.

13 Protokół rzutu monetą

13.1 Definicja

Protokoły służce do ustalania (losowania) wartości niezależnej od intencji użytkowników protokołów.

13.2 Cechy

- Achilles musi losować jakąś wartość, zanim Bachus zacznie odgadywać jej wartość
- Achilles nie może mieć możliwości dokonania ponownego losowania po usłyszeniu orzeczenia Bachusa
- Bachus nie może dowiedzieć się, co wylosował Achilles zanim podjął decyzję.

13.3 Przykład

Rzucanie monetą z wykorzystaniem funkcji jednokierunkowej.

1. Achilles wybiera losową liczbę x, oblicza y = f(x), gdzie f(x) jest funkcją jednokierunkową i przesyła wartość y do Bachusa.
2. Bachus odgaduje, czy x jest parzyste czy nieparzyste i przesyła swoje przypuszczenie do Achillesa

3. jeśli przypuszczenie Bachusa jest poprawne, to wynikiem jest „reszka”, jeżeli nieprawdziwe, to „orzeł”, Achilles przesyła rezultat do Bachusa
4. Bachus potwierdza, że y = f(x).

14 Ślepe podpisy cyfrowe

14.1 Definicja

Matematyczny sposób sprawdzenia autentyczności dokumentów i wiadomości elektronicznych. Poprawny podpis oznacza, że wiadomość pochodzi od właściwego nadawcy, który nie może zaprzeczyć faktowi jej nadania oraz że wiadomość nie została zmieniona podczas transmisji.

14.2 Cechy

- niepodrabialny
- niezaprzeczalny
- autentyczny
- zapewnia integralność dokumentu
- nie można go ponownie użyć
- może istnieć niezależnie
- rózny dla róznych dokumentów
- podpisujący **nie** zna dokumentu
- nie ma możliwości powiązania pary z wykonanym protokołem
- poprawne zakończenie protokołu, generuje zawsze pare wiadomość + cert

14.3 Przykład

Ślepe podpisy można zrealizować wykorzystując RSA.

Bachus posiada klucz jawny e, klucz prywatny d i moduł jawny n. Achilles chce by Bachus podpisał na ślepo wiadomość m.

1. Achilles wybiera losowo k z przedziału 1,n
2. Achilles zaciemnia m obliczając t = mk^e (mod n) i przesyła do Bachusa
3. Bachus podpisuje t: t^d = (m*k$^{e)}$d (mod n) i przesyła Achillesowi
4. Achilles usuwa zaciemnienie t^d poprzez obliczenie s = t^d/k (mod n)