TeleToken

Wersja 1.03

Spis treści

Założenia

Token jest zaprojektowany tak żeby w przypadku ujawnienia wszystkich dostępnych producentowi informacji na temat TeleTokenu, systemy zabezpieczone TeleTokenem nadal pozostały bezpieczne. Aby uzyskać takie bezpieczeństwo, żadna z informacji dostępnych dla producenta TeleTokenu nie może być kluczowa z punktu widzenia bezpieczeństwa TeleTokenu. Oznacza to, iż niemożliwe jest złamanie TeleTokenu nawet, jeśli dysponuje się pełną wiedzą na temat algorytmów, protokołów oraz haseł i identyfikatorów nadawanych w procesie produkcji, wstępnego programowania i niskiego formatowania. Z powyższego wynika, że wszystkie informacje kluczowe dla bezpieczeństwa TeleTokenu muszą być wpisywane przez odbiorcę TeleTokenu i muszą pozostać nie możliwe do odczytania.

Nie ma możliwości zabezpieczenia TeleTokenu przed przegraniem zawartości pamięci z jednego egzemplarza do innego, jednak po takiej operacji TeleToken, do którego przegrano „obcą" zawartość pamięci ma być nieprzydatny.

W celu zwiększenia bezpieczeństwa TeleTokenu, wszystkie zastosowane algorytmy kryptograficzne muszą być algorytmami powszechnie uznanymi za bezpieczne. Ponieważ trudno jest dowieść bezpieczeństwa algorytmu niejawnego przyjęto, że użyte algorytmy kryptograficzne muszą być jawne.

Zabezpieczenia przed złamaniem

Zabezpieczenie przed złamaniem TeleTokenu przy ujawnieniu protokołów i algorytmów możliwe jest tylko przez zastosowanie metod kryptograficznych. Oznacza to, iż wszystkie dane przechowywane w TeleTokenie jak i cała komunikacja z TeleTokenem musi być szyfrowana silnym algorytmem szyfrującym odpornym na złamanie. Przy takim rozwiązaniu algorytm szyfrowania może być jawny natomiast należy maksymalnie chronić klucze szyfrowania używane do komunikacji z TeleTokenem. W praktyce oznacza to, że klucze szyfrowania mogą być znane jedynie TeleTokenowi i programowi upoważnionemu do dostępu do TeleTokenu. Aby tak było nie może istnieć możliwość odczytania tych kluczy z TeleTokenu a klucze te mogą być znane i wpisywane do TeleTokenu wyłącznie przez odbiorcę TeleTokenu.

Zabezpieczenie przed przepisaniem pamięci pomiędzy TeleTokenami

Podczas wstępnego programowania TeleTokenu wpisywany jest do niego losowy i nieznany producentowi TeleTokenu klucz używany do szyfrowania danych w pamięci EEPROM. Klucz ten zapisany jest wyłącznie w pamięci procesora TeleTokenu i nie jest znana (również producentowi TeleTokenu) metoda odczytania go. Wszystkie dane przechowywane w pamięci zewnętrznej TeleTokenu są szyfrowane tym kluczem. Oznacza to, iż nie ma dostępu do rozszyfrowanych danych, a dane w postaci zaszyfrowanej są nieprzydatne z powodu braku możliwości ich odszyfrowania w rozsądnym czasie.

Założenia dodatkowe

Do jednego komputera może być równocześnie dołączone wiele TeleTokenów należących zarówno do jednego odbiorcy TeleTokenów jak i do różnych odbiorców. Równocześnie pojedynczy odbiorca może używać TeleTokenów do zabezpieczania różnych systemów. Przy takim założeniu w celu odnalezienia właściwego TeleTokenu oprogramowanie musiałoby odczytać wszystkie TeleTokeny dołączone do komputera a następnie zignorować te, które udzielą niepoprawnych odpowiedzi. Oznacza to konieczność skorzystania z haseł dostępu do TeleTokenu w momencie, w którym jeszcze nie wiadomo czy dostęp odbywa się do właściwego TeleTokenu. Aby tego uniknąć konieczna jest możliwość wstępnej selekcji TeleTokenów należących do poszczególnych odbiorców. W tym celu wprowadzono 4-bajtową liczbę identyfikującą odbiorcę. Identyfikator odbiorcy używany jest wyłącznie do wstępnej selekcji TeleTokenów i w oparciu o niego nie może być budowany żaden mechanizm zabezpieczeń. Identyfikator odbiorcy może być sprawdzony przy pomocy specjalnego uproszczonego protokołu.
Identyfikator odbiorcy zbudowany jest z dwóch członów:

- Najstarsze 3-bajty są przyznawane i nadawane przez producenta TeleTokenu i służą do jednoznacznej identyfikacji odbiorcy TeleTokenu. Wartość ta ustalana jest przez producenta w celu uniknięcia kolizji identyfikatorów pomiędzy odbiorcami.
- Najmłodszy bajt jest przeznaczony do wykorzystania przez odbiorcę TeleTokenów. Może być użyty np.: do wyszukiwania TeleTokenów posiadających specjalne uprawnienia w systemie. Jeżeli dany odbiorca używa TeleTokenów do zabezpieczenia kilku różnych produkowanych przez siebie systemów to bajt ten może być użyty do szybkiego odszukania TeleTokenu zabezpieczającego dany system.

Identyfikator odbiorcy jest wpisywany do TeleTokenu przez producenta podczas nisko poziomowego formatowania TeleTokenu i później jest niemożliwy do zmienienia (z wyjątkiem najmłodszego bajta).
Odbiorca otrzymuje TeleToken bez zaprogramowanych kluczy szyfrowania i musi je zapisać samodzielnie. Ponieważ klucze te nie są znane nikomu poza odbiorcą TeleTokenów, w przypadku ich utraty (spowodowanej np. jakimś błędem popełnionym przez odbiorcę) TeleToken staje się bezużyteczny. W takiej sytuacji przydatna była by możliwość odzyskania kontroli nad TeleTokenem przez odbiorcę. Ze względów bezpieczeństwa niedopuszczalne jest umożliwienie odczytu haseł z TeleTokenu. Również wprowadzenie możliwości dodawania haseł do TeleTokenu bez znajomości odpowiednich haseł zapisanych przez odbiorcę w TeleTokenie jest niedopuszczalna ze względów bezpieczeństwa. Jedynym akceptowalnym rozwiązaniem jest wprowadzenie możliwości doprowadzenia TeleTokenu do stanu „fabrycznego”, (czyli cała pamięć poza identyfikatorem odbiorcy i hasłem formatowania musi zostać wyczyszczona). Takie rozwiązanie umożliwia odzyskanie przez odbiorcę kontroli nad TeleTokenem a wszystkie dane zapisane w TeleTokenie zostaną bezpowrotnie stracone i nie zostaną ujawnione. Funkcjonalność taka może być przydatna również, gdy odbiorca z jakiś powodów będzie chciał w prosty i szybki sposób „zniszczyć” TeleToken. Odbiorca zawsze musi mieć możliwość „odzyskania” hasła formatowania TeleTokenu nawet, jeśli je utraci. Oznacza to, że hasło to musi być stałe (i różne dla różnych odbiorców) i być nadawane razem z identyfikatorem odbiorcy w procesie nisko poziomowego formatowania TeleTokenu. Hasło to pozostaje znane dla producenta TeleTokenu, dlatego też ze względów bezpieczeństwa nie może umożliwiać nic poza całkowitym (oprócz identyfikatora użytkownika i hasła formatowania) wyczyszczeniem pamięci TeleTokenu.

Podstawowa funkcjonalność TeleTokenu

Token posiada 4-bajtowy identyfikator odbiorcy umożliwiający wstępną selekcję TeleTokenów dołączonych do komputera. Trzy najstarsze bajty identyfikatora są programowane przez producenta a najmłodszy bajt pozostaje do wykorzystania przez odbiorcę.

Token posiada hasło umożliwiające jego sformatowanie przez odbiorcę. Po sformatowaniu cała pamięć przeznaczona dla użytkownika jest wyczyszczona i jest możliwość zapisania hasła początkowego.

Token posiada pamięć nieulotną na dane użytkownika. Pamięć ta podzielona jest na sektory po 16 bajtów (128 bitów).

Pewna ilość sektorów pamięci na dane użytkownika może być zabezpieczona przed ponownym zapisem. Zdjęcie zabezpieczenia możliwe jest tylko przez format TeleTokenu.
Token posiada pamięć ulotną na dane użytkownika. Pamięć ta podzielona jest na sektory po 16 bajtów (128 bitów).

Token posiada pamięć na tablice haseł. Tablica ta może być wyłącznie zapisywana, nie istnieje możliwość odczytania tej tablicy. Do każdego z haseł przyporządkowane są bity uprawnień danego hasła. Hasła z tej tablicy używane są do szyfrowania komunikacji, szyfrowania bloków danych i innych celów.

Token umożliwia zaszyfrowanie oraz rozszyfrowanie bloku 16-bajtowego hasłem z tablicy
Token posiada uniwersalne nieulotne 16-bitowe liczniki do wykorzystania przez odbiorcę, liczniki mogą zostać powiązane z pamięcią lub hasłem z tablicy

Token potrafi zwrócić informacje o rozmiarze pamięci, rozmiarze tablicy haseł, ilości liczników uniwersalnych i inne informacje o udostępnianych funkcjach.

Pamięć nieulotna na dane

Pamięć na dane jest pamięcią nieulotną i może być wykorzystana przez odbiorcę do przechowywania dowolnych własnych informacji. Jest ona podzielona na sektory po 16 bajtów (128 bitów), co jest zgodne z przyjętym rozmiarem bloku szyfrowania. Pamięć adresowana jest poprzez podanie numeru sektora. Operacje na pamięci (zapis i odczyt) możliwe są wyłącznie na całym sektorze. Numer sektora jest, jednobajtowy co umożliwia zaadresowanie 256 sektorów, czyli 4096 bajtów pamięci. Rozmiar pamięci dostępnej dla odbiorcy zależy od wersji TeleTokenu i może być z niego odczytany.

Zależnie od wersji TeleTokenu początkowe sektory pamięci mogą być niezależnie od siebie zabezpieczone przed zapisem. Zabezpieczenie przed zapisem dla poszczególnych sektorów da się wyłącznie ustawić. Jedyną możliwością odblokowania zapisu jest sformatowanie TeleTokenu, które kasuje wszystkie blokady zapisu do pamięci.
Pamięć ta posiada ograniczenie na ilość możliwych zapisów.

Pamięć ulotna na dane

Pamięć ta jest pamięcią ulotną jednak udostępnia ona identyczną funkcjonalność jak pamięć nieulotna z ta różnicą, że zawartość tej pamięci jest tracona przy restarcie oraz odłączeniu zasilania TeleTokenu. Każdy sektor pamięci ulotnej jest skojarzony z sektorem pamięci nieulotnej o tym samym numerze. Podczas startu TeleToken przepisuje zawartość sektorów pamięci nieulotnej do skojarzonych z nimi sektorów pamięci ulotnej. TeleToken nie dysponuje możliwością

bezpośredniego przepisania zawartości pamięci ulotnej do nieulotnej. Pamięć ta jest przeznaczona do przechowywania informacji, które mają zostać utracone przy restarcie lub wyłączeniu zasilania TeleTokenu. Pamięć ta nie może zostać zabezpieczona przed zapisem.

Pamięć ta nie posiada ograniczenia na ilość możliwych zapisów.

Hasła użytkownika

Token wyposażony jest w tablice służącą do przechowywania haseł. Hasła do tablicy mogą być wyłącznie zapisywane. Nie ma możliwości odczytania hasła z Tablicy. Hasła zapisane w tablicy używane są do:

- Szyfrowania komunikacji z TeleTokenem
- Szyfrowania bloków danych
- Odczytu i przestawiania liczników
- Zapisu haseł do tablicy

Z każdym z haseł powiązane są bity uprawnień hasła definiujące, do jakich operacji hasło może zostać użyte a do jakich nie. W ramach bitów uprawnień istnieje również możliwość zablokowania możliwości modyfikowania hasła i niezależnie bitów uprawnień hasła. Zdjęcie tych dwóch blokad możliwe jest wyłącznie poprzez sformatowanie TeleTokenu.

Liczniki uniwersalne

Token jest wyposażony w 16-bitowe liczniki uniwersalne nie tracące swojej zawartości po restarcie lub wyłączeniu zasilania TeleTokenu.
Każdy licznik może być:

- Zapisany
- Odczytany
- Zwiększony i odczytany
- Zmniejszony i odczytany

Każda z powyższych możliwości może zostać zablokowana poprzez skasowanie odpowiedniego bitu konfiguracji danego licznika. Istnieje możliwość zablokowania możliwości zmiany bitów konfiguracji licznika. Zdjęcie takiej blokady następuje wyłącznie przy formacie TeleTokenu. Dodatkowo każdemu z liczników można włączyć blokadę samoczynnej modyfikacji zawartości licznika po osiągnięciu przez licznik zera (zarówno przy odliczaniu w górę jak i w dół).

Każdy licznik może być powiązany z odpowiadającym mu sektorem pamięci lub pozycją w tablicy haseł. Powiązanie odbywa się według zasady: n-ty licznik może zostać powiązany, z ·n-tym sektorem i n-tym hasłem np. zerowy licznik może zostać powiązany wyłącznie z zerową stroną i hasłem na zerowej pozycji w tablicy. Poza szyfrowaniem komunikacji w przypadku powiązania z hasłem dowiązanie odnosi się do szyfrowania i deszyfrowania bloku danych. W przypadku powiązania z pamięcią można wybrać, do których operacji (odczyt lub zapis) licznik ma być dowiązany. Dowiązanie powoduje, że wykonanie operacji (użycie hasła do szyfrowania lub rozszyfrowania) możliwe jest tylko, gdy licznik jest różny od zera a każde wykonanie operacji (użycie hasła do szyfrowania lub rozszyfrowania) powoduje samoczynne zmniejszenie licznika o jeden.

Każdy z liczników może pracować w trybie samoczynnego generowania kolejnych kodów uwierzytelniających. W trybie tym dostępny jest rozkaz, który wykonuje następujące kroki:

- zwiększenie wartości licznika o jeden
- pobranie aktualnej wartości licznika i zaszyfrowanie jej hasłem o podanym numerze
- odesłanie do PC otrzymanego w powyższym kroku bloku danych

Definicje wstępne

- Klucz – wartość 16-to bajtowa używana do szyfrowania danych.
- Hasło – wartość 16-to bajtowa przechowywana w TeleTokenie w tablicy haseł. Zależnie od potrzeb hasło może być używane do różnych celów, ale nie ma możliwości jego odczytania.
- Sektor – 16-to bajtowy blok pamięci TeleTokenu przeznaczonej do przechowywania dowolnych danych przez odbiorcę TeleTokenu. Sektor jest najmniejszym niepodzielnym obszarem pamięci TeleTokenu. Wszystkie operacje odczytu pamięci zwracają zawartość całego sektora a operacje zapisu modyfikują zawartość całego sektora.

Notacja i oznaczenia stosowane przy opisie protokołu i algorytmów:

C(Klucz,Dane)	- operacja zaszyfrowania „Dane” przy użyciu klucza „Klucz”
D(Klucz,Dane)	- operacja rozszyfrowania „Dane” przy użyciu klucza „Klucz”
CD(Klucz,Dane)	- operacja zaszyfrowania lub rozszyfrowania „Dane” przy użyciu klucza „Klucz”(zależnie od kontekstu)

Założenia dodatkowe:

- W przypadku liczb wielobajtowych przyjęto notacje z pierwszym bajtem najmniej znaczącym.
- Wszystkie rozkazy maja jednakowy rozmiar wiadomości
- Token posiada dobry kryptograficznie generator liczb pseudolosowych

Protokół komunikacyjny

Protokół zapewnia kryptograficzną ochronę dostępu do TeleTokenu. Pojedyncza operacja na TeleTokenie zwana jest „sesją” i ma następujący przebieg:

- PC wysyła do TeleTokenu rozkaz rozpoczęcia sesji (KeyNr=0) proponując równocześnie klucz sesji (PassNrPC) oraz wartość losową będącą podstawą szyfrowania danych w sesji (RandPC)
- PC pobiera z TeleTokenu odpowiedz, w której TeleToken proponuje swój klucz sesji (PassNrT) oraz wartość losową będącą podstawą szyfrowania danych w sesji (RandT)
- PC wysyła do TeleTokenu zaszyfrowaną wiadomość zawierającą rozkaz wraz z parametrami i danymi
- PC pobiera z TeleTokenu odpowiedź zawierającą status wykonania rozkazu oraz opcjonalny wynik działania rozkazu
- PC wysyła do TeleTokenu potwierdzenie odebrania wyników oraz zakończenia sesji

Podczas proponowania kluczy sesji, klucze nie są przesyłanie wprost. Przesyłany jest jedynie numer pozycji w tablicy haseł. Wszystkie operacje kryptograficzne używane przez ten protokół oraz udostępniane przez TeleToken bazują na algorytmie „AES”(„Rijndael”).

Oznaczenia używane w opisie

H[n]	- operacja pobrania z tablicy haseł n-tego hasła
KeyNr	- Liczba używana do wyznaczenia hasała tymczasowego lub 0 – wiadomość nieszyfrowana
Cmd	- rozkaz do wykonania przez TeleToken
Status	- kod potwierdzający wykonanie rozkazu lub kod błędu
Addr	- numer sektora w pamięci TeleTokenu na którym ma zostać wykonana operacja odczytu/zapisu
PassNr	- Numer hasła w tablicy haseł, które ma zostać wykorzystane podczas operacji szyfrowania / deszyfrowania
Param	- parametr rozkazu przekazanego TeleTokenowi do wykonania, może to być np. PassNr lub Addr
PassNrPC	- numer hasła sesji zaproponowany przez PC
PassNrT	- numer hasła sesji zaproponowany przez TeleToken
RandPC	- wartość losowa do generowania klucza sesji zaproponowana przez PC
RandT	- wartość losowa do generowania klucza sesji zaproponowana przez TeleToken
Ks	- klucz sesji – klucz używany do szyfrowania danych w trakcie trwania pojedynczej sesji dostępu do TeleTokenu
Data	- blok przesyłanych danych
Control	- blok kontrolny

Algorytm szyfrowania bloków danych

Klucz sesji wyznaczany jest ze wzoru:

- Ks=C(H[PassNrPC],RandT) xor C(H[PassNrT],RandPC)

Należy zwrócić uwagę, że Rand zaproponowany przez PC jest szyfrowany hasłem zaproponowanym przez TeleToken a Rand zaproponowany przez TeleToken jest szyfrowany hasłem zaproponowanym przez PC.
Klucz sesji używany jest wyłącznie do generowania klucza bloku zgodnie ze wzorem:

- Kn=C(Ks,KeyNr|n|d) gdzie:
  n jest numerem szyfrowanego bloku w obrębie wiadomości
  d jest liczbą zależną od kierunku przepływu wiadomości i wynosi:
  0 dla PC -> TeleToken
  1 dla TeleToken -> PC

UWAGA! Należy zagwarantować, że kolejne wartości, KeyNr użyte przez PC w obrębie sesji będą rosnące! TeleToken ma obowiązek zakończyć sesję, jeśli wartość KeyNr otrzymana z PC jest mniejsza bądź równa od poprzedniej wartości KeyNr otrzymanej z PC w ramach tej samej sesji!

Blok danych jest szyfrowany kluczem bloku
Jeśli KeyNr jest wyzerowany to wiadomość jest niezaszyfrowana. Jeśli KeyNr jest różny od zera to cała wiadomość poza KeyNr jest zaszyfrowana zgodnie z podanym algorytmem.

Blok danych

Zawiera dane przekazywane z TeleTokenu do PC lub z PC do TeleTokenu. Rozmiar Tego bloku wynosi 16-bajtów i zajmuje cały zerowy blok wiadomości. (Bloki w wiadomości numerowane są od zera)

Blok rozkazu

Zawiera kod rozkazu oraz parametry potrzebne do wykonania rozkazu. Rozmiar tego bloku wynosi 4-bajty i zajmuje pierwsze 4-bajty pierwszego bloku wiadomości.

Blok kontrolny rozkazu

Aby sprawdzić poprawność komunikacji oraz rozszyfrowania danych każdy rozkaz zawiera blok kontrolny. Blok Ten ma rozmiar 12-bajtów i zajmuje ostatnie 12-bajtów pierwszego bloku danych wiadomości.
Blok ten jest wyznaczany według algorytmu:

- blok kontrolny wypełnić wartością „0xFF”
- kolejne bajty bloku „zxorować” z kolejnymi bajtami wiadomości (poza samym blokiem uwierzytelniającym). Ponieważ blok uwierzytelniający ma mniejszy rozmiar niż wiadomość to należy blok uwierzytelniający „zapętlić” (po ostatnim bajcie bloku występuje pierwszy bajt bloku)

UWAGA! Liczenie bloku uwierzytelniającego wykonać przed zaszyfrowaniem wiadomości!

UWAGA! Liczenie bloku kontrolnego jest wykonywane zawsze i nie jest zależne od wykonywanego rozkazu oraz od tego czy wiadomość jest szyfrowana czy nie!

Format wiadomości

Kolejność, rozmiar i nazwy pól w wiadomości przychodzącej i wychodzącej z TeleTokenu:

Rozmiar	***Nazwa pola:***	***Numer bloku danych***
3	KeyNr	nie szyfrowane
16	Data	0
1	Cmd/Status	1
1	Param1	1
1	Param2	1
1	Param3	1
12	Control	1

UWAGA! Jeśli KeyNr = 0 to wiadomość nie jest szyfrowana!
Dopuszczalne tylko dla wiadomości:

- formatowania TeleTokenu (PC -> TeleToken)
- rozpoczęcia sesji (PC->Token oraz TeleToken->PC)
- zakończenia sesji (tylko TeleToken->PC)

Przebieg pojedynczej sesji

W obrębie pojedynczej sesji wykonywane są następujące kroki:

1 Sesja rozpoczyna się od wysłania przez PC do TeleTokenu wiadomości otwarcia sesji.
W wiadomości tej pole KeyNr jest równe zero a w pole Cmd jest wpisany kod rozkazu otwarcia sesji.
Pole Param1 zawiera numer hasła (w tablicy haseł) proponowanego przez PC (czyli PassNrPC) a pole Data zawiera tablice wylosowaną przez PC (czyli RandPC).
Ostatnim polem wiadomości jest pole Control zawierające blok kontrolny.
Ze względów bezpieczeństwa protokołu tablica RandPC powinna być możliwie niepowtarzalna.

2 PC pobiera z TeleTokenu potwierdzenie otwarcia sesji.
W wiadomości tej pole Cmd/Status zawiera kod potwierdzenia otwarcia sesji
Pole Param1zawiera numer hasła (w tablicy haseł) proponowanego przez TeleToken (czyli PassNrT) a pole Data zawiera tablice losową wylosowaną przez TeleToken (czyli RandT). Ostatnim polem wiadomości jest pole Control zawierające blok kontrolny.

3 PC wysyła do TeleTokenu rozkaz do wykonania. Pole Cmd zawiera kod operacji do wykonania a pola Param zawierają parametry zależne od operacji do wykonania. Jeżeli operacja wymaga podania bloku danych to pole Data zawiera ten blok danych. Ostatnim polem wiadomości jest pole, Control zawierające blok kontrolny.

4 PC pobiera z TeleTokenu potwierdzenie wykonania rozkazu. W wiadomości tej pole Status zawiera kod zakończenia rozkazu (kod „sukcesu" lub kod błędu) a pole Data opcjonalne dane będące efektem działania rozkazu. Ostatnim polem wiadomości jest pole Control zawierające blok kontrolny.

5 PC wysyła do TeleTokenu rozkaz zamknięcia sesji. W wiadomości tej pole Cmd zawiera kod rozkazu zakończenia sesji. Ostatnim polem wiadomości jest pole Control zawierające blok kontrolny. Od momentu otrzymania tego rozkazu możliwe jest otwarcie nowej sesji.

Rozkazy udostępniane przez TeleToken

Rozkazy wykonywane przez TeleToken dzielą się na następujące kategorie:

- Rozkazy wykonywane wyłącznie w trakcie trwania sesji
  są to wszystkie standardowe rozkazy TeleTokenu wymagające przesłania do lub z TeleTokenu danych, haseł lub innych poufnych informacji,
  dodatkowo do rozkazów tych zalicza się rozkaz zakończenia sesji
- Rozkazy, które mogą być wydane wyłącznie przy nie otwartej sesji
  są to dwa rozkazy:
  - rozkaz otwarcia sesji
  - rozkaz sformatowania TeleTokenu – ten rozkaz jest używany między innymi w sytuacji utraty kontroli nad TeleTokenem czyli gdy nie da się rozpocząć sesji z TeleTokenem (np. w skutek utracenia haseł dostępu do TeleTokenu).

Kod rozkazu jest przekazywany w wiadomości w polu Cmd a ewentualne parametry w polach „Param”. Jeżeli rozkaz wymaga podania bloku danych to przekazywany on jest w polu Data. W odpowiedzi na rozkaz TeleToken zwraca wiadomość, w której w polu Status jest przekazywany status wykonania rozkazu a w polu Data blok danych zwracany przez rozkaz (o ile rozkaz zwraca blok danych, w przeciwnym przypadku jako dane zwracany jest ciąg zer).

Domyślnie wszystkie rozkazy mogą być wykonane tylko w ramach otwartej sesji chyba, że w opisie konkretnego rozkazu zaznaczono, że jest inaczej.
W wiadomościach przesyłanych z PC do TeleTokenu pole Cmd zawiera kod rozkazu do wykonania przez TeleToken.

W wiadomościach przesyłanych z TeleTokenu do PC pole Status zawiera informacje o wykonaniu ostatnio otrzymanego rozkazu oraz o stanie TeleTokenu.

W przypadku, gdy blok kontrolny nie jest poprawny sesja jest kończona.
Jeśli przy nie otwartej sesji zostanie pobrana odpowiedź z TeleTokenu to pole Status odpowiedzi będzie zawierać informacje o Statusie zakończenia sesji (informacja czy sesja została zakończona rozkazem zakończenia sesji czy z powodu przekroczenia czasu trwania sesji).

UWAGA! Pola wiadomości niewymienione w opisie rozkazu mają być wyzerowane! Wyjątkiem jest pole, KeyNr, które zostało opisane przy omawianiu algorytmy szyfrowania wiadomości. Format
Rozkaz ten może być użyty wyłącznie przy nie otwartej sesji!

Rozkaz ten służy do wyczyszczenia pamięci TeleTokenu i przywróceniu go do stanu, w jakim jest dostarczany przez producenta. Wykonanie rozkazu wymaga podania specjalnego hasła formatowania TeleTokenu.

Poszczególne pola wiadomości zawierają:

- Cmd - kod rozkazu sformatowania TeleTokenu
- Param1 - wartość, która zostanie zapisana w TeleTokenie jako najmłodszy bajt ID odbiorcy, TeleTokenu
- Data - pięć pierwszych bajtów to hasło formatowania TeleTokenu, pozostałe wyzerowane

UWAGA! W tym rozkazie wiadomość nie jest szyfrowana a KeyNr musi być równy zero!
Rozkaz ten nie zwraca żadnego statusu wykonania ani nie sygnalizuje jawnie zakończenia

formatowania. Po sformatowaniu TeleToken wykonuje „reboot”, co może, (ale nie musi)spowodować utratę komunikacji z TeleTokenem (również utrata komunikacji z TeleTokenem nie oznacza, że został on sformatowany). Zawartość całej pamięci użytkownika, tablica haseł oraz liczniki są czyszczone. Dokładny opis stanu sformatowanego TeleTokenu zamieszczony jest w rozdziale *„Stan TeleTokenu po sformatowaniu”*.

UWAGA! Ze względów bezpieczeństwa w przypadku tablicy haseł nie wolno zakładać, że w sformatowanym TeleTokenie wszystkie hasła są usunięte! W przypadku, gdy w tablicy haseł są nieużywane pola to należy samodzielnie ustawić ich bity konfiguracji tak żeby zablokować możliwość skorzystania z nich!

UWAGA! Nie ma zdefiniowanej metody sprawdzenia czy formatowanie zakończyło się poprawnie! Operacja formatowania TeleTokenu powinna być wykonywana tylko w uzasadnionych przypadkach a konkretny algorytm postępowania przed i po formatowaniu jest zależny od specyfiki wykorzystania TeleTokenu.

Otwarcie sesji

Rozkaz ten służy do rozpoczęcia sesji z TeleTokenem. Do TeleTokenu przesyłane są: numer proponowanego przez PC hasła oraz proponowana przez PC tablica losowa używana do szyfrowania danych podczas trwania sesji. W odpowiedzi TeleToken odsyła proponowany przez siebie: numer hasła oraz tablice losową do szyfrowania danych w obrębie sesji.

Poszczególne pola wiadomości z PC do TeleTokenu zawierają:

- Cmd - kod rozkazu otwarcia sesji
- Param1 - PassNrPC - numer hasła proponowanego przez PC
- Data - RandPC - tablica losowa proponowana przez PC

Poszczególne pola wiadomości z TeleTokenu do PC zawierają:

- Status - kod informujący o powodzeniu lub błędzie wykonania rozkazu
- Param1 - PassNrT - numer hasła proponowanego przez PC
- Data - RandT - tablica losowa proponowana przez TeleToken

Możliwe błędy:

- TVTTS_NO_OPEN - oznacza to, że hasło o podanym numerze nie może być zaproponowane przez PC do szyfrowania danych podczas komunikacji z TeleTokenem lub, że wystąpił inny błąd uniemożliwiający otwarcie sesji.

UWAGA! W tym rozkazie wiadomość nie jest szyfrowana a KeyNr musi być równy zero!

Zamknięcie sesji

Rozkaz służy do zakończenia sesji. Używany jest w celu wyczyszczenia buforów komunikacyjnych TeleTokenu oraz udostępnienia (bez oczekiwania na przekroczenie czasu trwania sesji) TeleTokenu innym procesom.

Poszczególne pola wiadomości z PC do TeleTokenu zawierają:

- Cmd - kod rozkazu zamknięcia sesji

Po wydaniu tego rozkazu nie należy pobierać odpowiedzi z TeleTokenu.

Odczyt pamięci

Rozkaz służy do odczytu zawartości pamięci TeleTokenu.
Możliwy jest wyłącznie odczyt jednego całego sektora pamięci (bloku 16-bajtowego), przy czym adres przekazywany w tym rozkazie jest numerem sektora do odczytania.

UWAGA! Rozkazowi temu przypisane są dwa kody liczbowe zależnie od tego czy odczyt ma nastąpić z pamięci ulotnej czy nieulotnej.

Poszczególne pola wiadomości z PC do TeleTokenu zawierają:

- Cmd - kod rozkazu odczytu pamięci
- Param1 - Addr - Numer sektora do odczytania

Poszczególne pola wiadomości z TeleTokenu do PC zawierają:

- Status - kod informujący o powodzeniu lub błędzie wykonania rozkazu
- Data - dane odczytane z pamięci TeleTokenu

W efekcie wykonania rozkazu TeleToken może w polu Statusu zwrócić następujące błędy:

- TVTTS_OUT_OF_RANGE - adres z poza zakresu - sektor o podanym adresie nie istnieje
- TVTTS_PASS_NOT_PERMITTED - operacja niedopuszczalna dla hasła zaproponowanego podczas otwarcia sesji - hasło zaproponowane przy otwarciu sesji nie może być użyte do wygenerowania klucza sesji podczas odczytu pamięci
- TVTTS_ACCESS_EXPIRED - przekroczono dopuszczalną ilość odczytów sektora

Zapis pamięci oraz ustanowienie ochrony pamięci przed zapisem

Rozkaz służy do zapisania bloku danych do pamięci TeleTokenu oraz umożliwia zablokowanie zapisu do wybranego sektora w pamięci.

UWAGA! Zależnie od użytego kodu rozkazu możliwe są następujące operacje:

- zapis do sektora pamięci ulotnej
- zapis do sektora pamięci nieulotnej
- zapis do sektora pamięci nieulotnej i zablokowanie możliwości zapisu do tego sektora
- zablokowanie możliwości zapisu do sektora (bez zapisu danych do tego sektora)

Możliwy jest wyłącznie zapis jednego całego sektora pamięci (bloku 16-bajtowego), przy czym adres przekazywany w tym rozkazie jest numerem sektora do zapisania.

Poszczególne pola wiadomości z PC do TeleTokenu zawierają:

- Cmd - kod rozkazu zapisu pamięci
- Param1 - Addr - Numer sektora do zapisania
- Data - dane do zapisania do pamięci TeleTokenu

UWAGA! Dla rozkazu służącego wyłącznie do zablokowania możliwości zapisu do sektora pole danych jest ignorowane i powinno być wyzerowane.

Poszczególne pola wiadomości z TeleTokenu do PC zawierają:

- Status - kod informujący o powodzeniu lub błędzie wykonania rozkazu

- W efekcie wykonania rozkazu TeleToken może w polu Statusu zwrócić następujące błędy:

- TVTTS_OUT_OF_RANGE - adres z poza zakresu - sektor o podanym adresie nie istnieje
- TVTTS_PROTECTION_NOT_SUPPORTED - sektor jest poza zakresem sektorów, dla których możliwa jest ochrona przed zapisem
- TVTTS_PASS_NOT_PERMITTED - operacja niedopuszczalna dla hasła zaproponowanego podczas otwarcia sesji - hasło zaproponowane przy otwarciu sesji nie może być użyte do wygenerowania klucza sesji podczas zapisu do pamięci.
- TVTTS_NOT_PERMITTED - sektor jest zabezpieczony przed zapisem
- TVTTS_ACCESS_EXPIRED - przekroczono dopuszczalną ilość zapisów do sektora
- VCTS_DAMAGED - błąd zapisu do pamięci - błąd krytyczny, powoduje zablokowanie TeleTokenu

Szyfrowanie i rozszyfrowywanie hasłem

Rozkaz służy do szyfrowania i rozszyfrowywania danych przy użyciu hasła zapisanego w tablicy haseł TeleTokenu.

UWAGA! Rozkazowi temu przypisane są dwa kody liczbowe zależnie od tego czy ma nastąpić zaszyfrowanie czy rozszyfrowanie bloku danych.

Poszczególne pola wiadomości z PC do TeleTokenu zawierają:

- Cmd - kod rozkazu szyfrowania lub rozszyfrowania
- Param1 - PassNr - numer w tablicy haseł TeleTokenu hasła, które ma zostać użyte do (roz)szyfrowania bloku danych
- Data - blok danych, który ma zostać zaszyfrowany lub rozszyfrowany przez TeleToken

Poszczególne pola wiadomości TeleTokenu do PC zawierają:

- Status - kod informujący o powodzeniu lub błędzie wykonania rozkazu
- Data - blok danych zaszyfrowanych lub rozszyfrowanych przez TeleToken

W efekcie wykonania rozkazu TeleToken może w polu Statusu zwrócić następujące błędy:

- TVTTS_PASS_NOT_PERMITTED - operacja niedopuszczalna dla hasła zaproponowanego podczas otwarcia sesji - hasło zaproponowane przy otwarciu sesji nie może być użyte do wygenerowania klucza sesji podczas szyfrowania lub rozszyfrowywania bloku danych
- TVTTS_NOT_PERMITTED - hasło o podanym numerze nie może zostać użyte do szyfrowania lub rozszyfrowywania danych
- TVTTS_ACCESS_EXPIRED - przekroczono dopuszczalną ilość użyć hasła do (roz)szyfrowania

Odczyt i modyfikacja liczników

Rozkaz służy do odczytania zawartości licznika uniwersalnego. Przed odczytem może nastąpić samoczynne zmniejszenie lub zwiększenie wartości licznika.

UWAGA! Zależnie od użytego kodu rozkazu możliwe są następujące operacje:

- odczytanie licznika
- zwiększenie o jeden i odczytanie licznika
- zmniejszenie o jeden i odczytanie licznika
- zwiększenie o jeden i odesłanie zaszyfrowanej wartości licznika

Poszczególne pola wiadomości z PC do TeleTokenu zawierają:

- Cmd - kod rozkazu odczytu licznika
- Param1 - CntNr - numer licznika
- Param2 – PassNr – numer hasła do zaszyfrowania wartości pobranej z licznika

Dla normalnego odczytu pole Param2 jest wyzerowane
Dla odczytu szyfrowanego pole Param2 zawiera numer hasła, którym ma zostać zaszyfrowana przed przesłaniem do PC wartość odczytana z licznika.

Poszczególne pola wiadomości TeleTokenu do PC zawierają:

- Status - kod informujący o powodzeniu lub błędzie wykonania rozkazu
- Data - wartość odczytana z licznika

Dla normalnego odczytu w dwóch pierwszych bajtach w polu Data jest zwracana wartość licznika w kolejności Młodszy bajt / Starszy bajt.

Dla odczytu szyfrowanego pole Data zawiera całe 16-bajtów będących efektem szyfrowania.
W efekcie wykonania rozkazu TeleToken może w polu Statusu zwrócić następujące błędy:

- TVTTS_OUT_OF_RANGE - numer licznika z poza zakresu - licznik o podanym adresie nie istnieje
- TVTTS_PASS_NOT_PERMITTED - operacja niedopuszczalna dla hasła zaproponowanego podczas otwarcia sesji - hasło zaproponowane przy otwarciu sesji nie może być użyte do wygenerowania klucza sesji podczas odczytu licznika
- TVTTS_NOT_PERMITTED - licznik nie ma odblokowanej możliwości wykonania podanego rozkazu
- TVTTS_DAMAGED - błąd zapisu do licznika - błąd krytyczny, powoduje zablokowanie TeleTokenu
- TVTTS_ACCESS_EXPIRED - licznik osiągnął wartość zero i został zatrzymany

Zapis do liczników

Rozkaz służy do zapisu danych do licznika uniwersalnego.
Poszczególne pola wiadomości z PC do TeleTokenu zawierają:

- Cmd - kod rozkazu odczytu licznika
- Param1 - CntNr - numer licznika
- Data - dwa pierwsze bajty zawierają wartość, która ma zostać wpisana do licznika (Młodszy bajt / Starszy bajt)

Poszczególne pola wiadomości z TeleTokenu do PC zawierają:

- Status - kod informujący o powodzeniu lub błędzie wykonania rozkazu

W efekcie wykonania rozkazu TeleToken może w polu Statusu zwrócić następujące błędy:

- TVTTS_OUT_OF_RANGE - numer licznika z poza zakresu - licznik o podanym adresie nie istnieje
- TVTTS_PASS_NOT_PERMITTED - operacja niedopuszczalna dla hasła zaproponowanego podczas otwarcia sesji - hasło zaproponowane przy otwarciu sesji nie może być użyte do wygenerowania klucza sesji podczas zapisu do licznika
- TVTTS_NOT_PERMITTED - Licznik nie ma odblokowanej możliwości wykonania podanego rozkazu
- TVTTS_DAMAGED - błąd zapisu do licznika - błąd krytyczny, powoduje zablokowanie TeleTokenu

Zapis do obszaru konfiguracji liczników

Rozkaz służy do modyfikowania stanu bitów konfiguracji licznika uniwersalnego.

UWAGA! Zależnie od użytego kodu rozkazu możliwe są następujące operacje:
- wpisanie podanych stanów do bitów konfiguracji
- ustawienie podanych bitów konfiguracji
- skasowanie podanych bitów konfiguracji

Poszczególne pola wiadomości z PC do TeleTokenu zawierają:
- Cmd - kod rozkazu odczytu licznika
- Param1 - CntNr - numer licznika
- Data - dwa pierwsze bajty zawierają maskę bitową lub stan bitów konfiguracji

Dla rozkazu zapisu bitów wartość przekazana w Data jest wprost wpisywana do obszaru bitów konfiguracji licznika.

Dla rozkazu ustawienia bitów konfiguracji wartość przekazana w Data jest maską bitową bitów do ustawienia (zostaną ustawione te bity, które w masce są ustawione a stan pozostałych bitów nie ulegnie zmianie)

Dla rozkazu skasowania bitów konfiguracji wartość przekazana w Data jest maską bitową bitów do skasowania (zostaną skasowane te bity, które w masce są ustawione a stan pozostałych bitów nie ulegnie zmianie)

Poszczególne pola wiadomości z TeleTokenu do PC zawierają:
- Status - kod informujący o powodzeniu lub błędzie wykonania rozkazu

W efekcie wykonania rozkazu TeleToken może w polu Statusu zwrócić następujące błędy:
- TVTTS_OUT_OF_RANGE - numer licznika z poza zakresu - licznik o podanym adresie nie istnieje
- TVTTS_PASS_NOT_PERMITTED - operacja niedopuszczalna dla hasła zaproponowanego podczas otwarcia sesji - hasło zaproponowane przy otwarciu sesji nie może być użyte do wygenerowania klucza sesji podczas zapisu do bitów konfiguracji licznika
- TVTTS_NOT_PERMITTED - licznik nie ma odblokowanej możliwości wykonania podanego rozkazu
- TVTTS_DAMAGED - błąd zapisu do licznika - błąd krytyczny, powoduje zablokowanie TeleTokenu

Bity konfiguracji licznika:
- Licznik może być zapisywany
- Licznik może być odczytany bez modyfikacji
- Licznik może być jednym rozkazem zwiększony i odczytany
- Licznik może być jednym rozkazem zmniejszony i odczytany
- Licznik może być jednym rozkazem zwiększony i odczytany a odczytana wartość ma zostać przed przesłaniem zaszyfrowana hasłem o podanym numerze
- Licznik ma być zatrzymany po dojściu do zera
- Licznik jest powiązany z odczytem pamięci
- Licznik jest powiązany z zapisem pamięci
- Licznik jest powiązany z hasłem w tablicy
- Odblokowanie możliwości modyfikowania bitów konfiguracji licznika (UWAGA! Ten bit można tylko skasować!).

Zapis do tablicy haseł

Rozkaz służy do zapisu nowego hasła do tablicy haseł.

Poszczególne pola wiadomości z PC do TeleTokenu zawierają:

- Cmd - kod rozkazu zapisu do tablicy haseł
- Param1 - PassNr - numer pozycji, do której ma nastąpić zapis w tablicy haseł
- Data - hasło do zapisania do tablicy

Poszczególne pola wiadomości z TeleTokenu do PC zawierają:

- Status - kod informujący o powodzeniu lub błędzie wykonania rozkazu

W efekcie wykonania rozkazu TeleToken może w polu Statusu zwrócić następujące błędy:

- TVTTS_OUT_OF_RANGE - numer hasła z poza zakresu - pozycja o podanym numerze nie istnieje w tablicy.
- TVTTS_PASS_NOT_PERMITTED - operacja niedopuszczalna dla hasła zaproponowanego podczas otwarcia sesji - hasło zaproponowane przy otwarciu sesji nie może być użyte do wygenerowania klucza sesji podczas zapisu tablicy haseł.
- TVTTS_NOT_PERMITTED - podana pozycja w tablicy ma zablokowaną możliwość zapisu hasła
- TVTTS_DAMAGED - błąd zapisu do tablicy - błąd krytyczny, powoduje zablokowanie TeleTokenu

Zapis bitów konfiguracji haseł

Rozkaz służy do modyfikowania stanu bitów konfiguracji hasła w tablicy.
UWAGA! Zależnie od użytego kodu rozkazu możliwe są następujące operacje:

- wpisanie podanych stanów do bitów konfiguracji
- ustawienie podanych bitów konfiguracji
- skasowanie podanych bitów konfiguracji

Poszczególne pola wiadomości z PC do TeleTokenu zawierają:

- Cmd - kod rozkazu odczytu licznika
- Param1 - PassNr - numer hasła w tablicy
- Data - dwa pierwsze bajty zawierają maskę bitową lub stan bitów konfiguracji

Dla rozkazu zapisu bitów wartość przekazana w Data jest wprost wpisywana do obszaru bitów konfiguracji hasła.

Dla rozkazu ustawienia bitów konfiguracji wartość przekazana w Data jest maską bitową bitów do ustawienia (zostaną ustawione te bity, które w masce są ustawione a stan pozostałych bitów nie ulegnie zmianie)

Dla rozkazu skasowania bitów konfiguracji wartość przekazana w Data jest maską bitową bitów do skasowania (zostaną skasowane te bity, które w masce są ustawione a stan pozostałych bitów nie ulegnie zmianie)

Poszczególne pola wiadomości z TeleTokenu do PC zawierają:

- Status - kod informujący o powodzeniu lub błędzie wykonania rozkazu

W efekcie wykonania rozkazu TeleToken może w polu Statusu zwrócić następujące błędy:

- TVTTS_OUT_OF_RANGE - numer hasła z poza zakresu - pozycja o podanym numerze nie istnieje w tablicy
- TVTTS_PASS_NOT_PERMITTED - operacja niedopuszczalna dla hasła zaproponowanego podczas otwarcia sesji - hasło zaproponowane przy otwarciu sesji nie może być użyte do wygenerowania klucza sesji podczas zapisu do bitów konfiguracji hasła.
- TVTTS_NOT_PERMITTED - podana pozycja w tablicy ma zablokowaną możliwość zapisu bitów konfiguracji hasła.
- TVTTS_DAMAGED - błąd zapisu do tablicy - błąd krytyczny, powoduje zablokowanie TeleTokenu.

Bity konfiguracji hasła:

- hasło może być proponowane przez TeleToken do generowania klucza sesji
- hasło może być proponowane przez PC do generowania klucza sesji wpisania hasła
- hasło może być proponowane przez PC do generowania klucza sesji modyfikacji bitów konfiguracji hasła
- hasło może być proponowane przez PC do generowania klucza sesji odczytu pamięci
- hasło może być proponowane przez PC do generowania klucza sesji zapisu pamięci
- hasło może być proponowane przez PC do generowania klucza sesji szyfrowania
- hasło może być proponowane przez PC do generowania klucza sesji deszyfrowania
- hasło może być proponowane przez PC do generowania klucza sesji odczytu licznika
- hasło może być proponowane przez PC do generowania klucza sesji zapisu licznika
- hasło może być proponowane przez PC do generowania klucza sesji modyfikacji bitów konfiguracji licznika

- hasło może być używane do szyfrowania
- hasło może być używane do rozszyfrowywania
- hasło może być używane do szyfrowania wartości odczytanej z licznika (praca w trybie generacji haseł uwierzytelniających)
- odblokowanie możliwości nadpisania hasła (UWAGA! Ten bit można tylko skasować!)
- odblokowanie możliwości modyfikowania bitów konfiguracji hasła (UWAGA! Ten bit można tylko skasować!).

Standardowe błędy zwracane przez TeleToken w polu „Status”

W odpowiedzi na wszystkie rozkazy TeleToken może zwrócić następujące kody błędów:

- TVTTS_NO_OPEN – sesja nie została otwarta a rozkaz wymagał otwartej sesji, przez co nie został wykonany (wyjątkiem jest rozkaz otwarcia sesji)
- TVTTS_EXPIRED - oznacza, że upłynął czas przeznaczony na sesję i sesja została zakończona. Nie jest zdefiniowane czy rozkaz został wykonany czy nie.
- TVTTS_NOT_SUPPORTED – podany kod rozkazu jest nieznany lub nie zaimplementowany i TeleToken nie mógł wykonać rozkazu.
- TVTTS_DAMAGED - oznacza, że TeleToken uległ uszkodzeniu i nie będzie akceptował żadnych rozkazów.

Funkcje biblioteki

TVTT_GetVersion

Funkcja przeznaczona do odczytu: wersji, podwersji oraz aktualnego API biblioteki.

Deklaracji funkcji:
DWORD WINAPI TVTT_GetVersion(DWORD * Ver, DWORD * SubVer, DWORD *APIVer)

Parametry:
- wskaźnik na DWORD – adres pod który zostanie zapisana wersja biblioteki
- wskaźnik na DWORD – adres pod który zostanie zapisana podwersja biblioteki
- wskaźnik na DWORD – adres pod który zostanie zapisany typ API (ulega zmianie w przypadku zmiany parametrów przekazywanych z i do funkcji)

TVTT_FindToken

Funkcja sprawdza czy do portu USB komputera został podpięty TeleToken o podanym identyfikatorze odbiorcy. Jeżeli został odnaleziony, funkcja zwraca uchwyt urządzenia, ważny do chwili zamknięcia komunikacji z TeleTokenem lub odłączenia go od portu USB.

Deklaracji funkcji:
DWORD WINAPI TVTT_FindToken(DWORD *TokenH, DWORD TimeOut, DWORD OwnerID,
BYTE OwnerMsk, BYTE Skip, DWORD * FeatureTblSize,
DWORD * FeatureTbl)

Parametry:
- wskaźnik na DWORD – adres pod który zostanie zapisany uchwyt
- DWORD – wartość jednostek czasu przeznaczonego na wykonanie funkcji (wartość 1 równa jest 1/10 sekundy)
- DWORD – identyfikator odbiorcy
- BYTE – maska dla ostatniego bajtu identyfikatora odbiorcy
- BYTE – liczba TeleTokenów do pominięcia podczas przeszukiwaniu portów USB
- wskaźnik na DWORD – rozmiar tablicy przekazywanej jako kolejny parametr funkcji. Przy zakończeniu wywołania funkcja zwraca informację ile pól zostało zapisanych.
- wskaźnik na tablice DWORD – funkcja zapisuje w tablicy kolejno następujące informacje:
    - znaleziony numer identyfikatora odbiorcy
    - ilość sektorów pamięci nieulotnej
    - ilość sektorów pamięci ulotnej
    - ilość początkowych sektorów pamięci nieulotnej, które można zabezpieczyć przed zapisem
    - ilość pozycji w tablicy haseł
    - ilość bitów konfiguracji przypadających na hasło
    - liczba liczników uniwersalnych
    - liczba bitów konfiguracji na licznik

Uwagi:
Maska ostatniego bajtu identyfikatora odbiorcy umożliwia ustalenie, które bity tego bajtu mają znaczenie podczas wyszukiwania TeleTokenu. Ustawienie bitu w masce na jeden oznacza, że odpowiadające mu bity identyfikatora w TeleTokenie i identyfikatora przekazanego do funkcji muszą być identyczne.

TVTT_FreeToken

Funkcja zamyka dostęp do TeleTokenu i zwalnia zajęte zasoby.

Deklaracja funkcji:
DWORD WINAPI TVTT_FreeToken(DWORD TeleTokenH)

Parametry:
- DWORD – uchwyt na TeleToken

TVTT_StartSession

Funkcja otwiera sesję komunikacji z TeleTokenem.

Deklaracja funkcji:
DWORD WINAPI TVTT_StartSession(DWORD TeleTokenH, DWORD TimeOut)

Parametry:
- DWORD – uchwyt na TeleToken
- DWORD – wartość jednostek czas przeznaczonego na wykonanie funkcji (wartość 1 równa jest 1/10 sekundy)

TVTT_Talk

Funkcja komunikacji z TeleTokenem. Przed pierwszym wywołaniem wymaga otwarcia sesji.

Deklaracja:
DWORD WINAPI TVTT_Talk(DWORD TeleTokenH, DWORD TimeOut, DWORD OutKey,
BYTE *OutDataA, BYTE *OutDataB, DWORD *InKey,
BYTE *InDataA, BYTE *InDataB)

Parametry:
- DWORD – uchwyt na TeleToken
- DWORD – wartość jednostek czasu przeznaczonego na wykonanie funkcji (wartość 1 równa jest 1/10 sekundy)
- DWORD – wektor inicjujący szyfrowanie dla wiadomości wychodzącej
- wskaźnik na tablice BYTE – zerowy blok danych wychodzących (1 część wiadomości)
- wskaźnik na tablice BYTE – pierwszy blok danych wychodzących (2 część wiadomości)
- wskaźnik na DWORD – wektor inicjujący szyfrowanie dla wiadomości przychodzącej
- wskaźnik na tablice BYTE – zerowy blok danych przychodzących (1 część wiadomości)
- wskaźnik na tablice BYTE – pierwszy blok danych przychodzących (2 część wiadomości)

TVTT_EndSession

Funkcja kończy sesję.

Deklaracja:
DWORD WINAPI TVTT_EndSession(DWORD TeleTokenH)

Parametry:
- DWORD – uchwyt na TeleToken

TVTT_Calc

Funkcja liczy blok kontrolny całej wiadomości a wynik umieszcza w właściwym miejscu wiadomości.

Deklaracja:
void TVTT_Calc(BYTE *DataA, BYTE*DataB)

Parametry:
- Adres na pierwszy blok (16-bajtów) wiadomości
- Adres na drugi blok (16-bajtów) wiadomości

Zwracane wartości:
- Wyliczony blok kontrolny wiadomości wpisany we właściwe miejsce wiadomości

Uwagi:
Każda wiadomość wysyłana do TeleTokenu musi mieć poprawny blok kontrolny niezależnie od tego czy wiadomość jest szyfrowana czy nie.

TVTT_Check

BYTE TVTT_Check(BYTE *DataA, BYTE*DataB)

Funkcja sprawdza poprawność bloku kontrolnego w wiadomości.

Parametry:
- Adres na pierwszy blok (16-bajtów) wiadomości
- Adres na drugi blok (16-bajtów) wiadomości

Zwracane wartości:
- Zero, jeśli wiadomość zawiera poprawny blok kontrolny
- Nie zero, jeśli blok kontrolny jest niepoprawny

Poprawność wiadomości należy sprawdzać dla każdej odebranej wiadomości niezależnie od tego czy wiadomości są szyfrowane. Niezgodność bloku kontrolnego należy potraktować jako poważny błąd komunikacji z TeleTokenem. Jedynym wyjątkiem od tej reguły jest sytuacja, w której otrzymano wiadomość zaszyfrowaną a oczekiwano wiadomości niezaszyfrowanej. Taka sytuacja jest sama w sobie poważnym błędem komunikacji a dalsza analiza odebranych danych jest bezcelowa.

TVTT_CalcSessionKey

Funkcja wylicza klucz sesji

Deklaracja:
void TVTT_CalcSessionKey (BYTE SessionKey[16], BYTE PassPC[16],
BYTE RandPC[16], BYTE PassT[16], BYTE RandT[16])

Parametry:
- Adres, pod który funkcja ma wpisać wyliczony klucz sesji
- Hasło PC
- Tablica losowa PC
- Hasło TeleTokenu
- Tablica losowa TeleTokenu

Zwracane wartości:
- Wyliczony klucz sesji wpisany do podanej tablicy

Uwagi:
Hasło PC jest hasłem, którego numer został przesłany do TeleTokenu w rozkazie rozpoczęcia sesji.
Hasło TeleTokenu jest hasłem, którego numer został przesłany z TeleTokenu w rozkazie rozpoczęcia sesji.
Tablica losowa PC to tablica, która została wysłana do TeleTokenu w rozkazie rozpoczęcia sesji.
Tablica losowa TeleTokenu to tablica, która została przysłana z TeleTokenu w rozkazie rozpoczęcia sesji.

Wyliczony klucz sesji należy zachować do czasu zakończenia sesji a następnie należy go bezwzględnie zamazać. Pozostałe wartości przekazane funkcji należy bezwzględnie zamazać od razu po wywołaniu tej funkcji.
Należy pamiętać, że bezpieczeństwo całego systemu jest ograniczone bezpieczeństwem przechowywania haseł dostępu do TeleTokenu.

TVTT_CryptBlock

Funkcja rozszyfrowuje lub szyfruje podany blok danych (zależnie od podanych parametrów)

Deklaracja
void TVTT_CryptBlock(BYTE Block[16], DWORD Key, BYTE Pass[16],
BYTE BlockNr, BYTE Direction)

Parametry:
- Adres na blok do zaszyfrowania (lub rozszyfrowania)
- Klucza szyfrowania (lub rozszyfrowywania) wiadomości
- Hasło sesji
- Numer bloku danych w obrębie wiadomości
- Kierunek przesyłania wiadomości

Zwracane wartości:
- Zaszyfrowany (lub rozszyfrowany) blok danych

Uwagi:
Szyfrowanie (i rozszyfrowywanie) należy przeprowadzać oddzielnie dla każdego bloku wiadomości.
Klucz szyfrowania wiadomości jest to wartość wektora inicjującego szyfrowanie dla wiadomości wychodzącej w funkcji TVTT_Talk.
Klucz rozszyfrowywania wiadomości jest to wartość wektora inicjującego szyfrowanie dla wiadomości przychodzącej w funkcji TVTT_Talk
Numer bloku danych w obrębie wiadomości jest zdefiniowany jako stałe:

- TVTT_BLOCK_A (wartość 0) oznacza szyfrowanie lub rozszyfrowywanie pierwszego bloku wiadomości
- TVTT_BLOCK_B (wartość 1) oznacza szyfrowanie lub rozszyfrowywanie drugiego bloku wiadomości

Kierunek przesyłania wiadomości definiuje czy wiadomość ma zostać zaszyfrowana czy rozszyfrowana i jest zdefiniowany jako stałe

- TVTT_PC2T (wartość 0) oznacza szyfrowanie wiadomości.
- TVTT_T2PC (wartość 1) oznacza rozszyfrowywanie wiadomości.

TVTT_Randomize

Funkcja dodaje zdarzenia losowe do generatora liczb pseudolosowych

Deklaracja:
void TVTT_Randomize(BYTE SrcNr,DWORD Data)

Parametry:

- Numer źródła zdarzeń losowych
- Wartość zdarzenia

Uwagi:
Funkcja dopuszcza maksymalnie 32 źródła zdarzeń losowych. Jeżeli dostępne jest więcej źródeł to należy je połączyć w grupy i jako numer źródła podawać numer grupy źródeł.
Jako wartość zdarzenia można użyć każdej wartości losowej związanej ze zdarzeniem, przy czym dla każdej z tych wartości można użyć innego numeru źródła zdarzeń losowych (np. dla zdarzenia naciśnięcia klawisza dostępne są dwie wartości: kod naciśniętego klawisza i czas systemowy w momencie naciśnięcia klawisza i każda z tych wartości może być użyta z innym numerem źródła zdarzeń losowych).
Dobrym rozwiązaniem jest podpięcie tej funkcji do procedury obsługi wiadomości systemowych.

TVTT_Rand

Funkcja generuje 16-bajtowy blok pseudolosowy.

Deklaracja:
void TVTT_Rand(BYTE Rand[16])

Parametry:

- Adres na tablice do wypełnienia wartościami losowymi

Zwracane wartości:

- Blok danych pseudolosowych

TVTT_RandGetInit

Funkcja zwraca blok danych do użycia przy następnej inicjalizacji generatora liczb pseudolosowych.

Deklaracja:
void TVTT_RandGetInit(BYTE Init[16*4])

Parametry:

- Adres na tablicę, do której nastąpi zapis danych opisujących stan generatora

Zwracane wartości:

- Blok danych opisujących stan generatora

Funkcję tę należy wywołać zaraz po zainicjowaniu generatora liczb pseudolosowych oraz przy wyjściu z programu, a zwróconą przez nią tablicę należy zachować w bezpiecznym miejscu i użyć przy następnym uruchomieniu programu.
Dane zwrócone przez ta funkcję należy przechowywać tak, aby możliwie utrudnić ich zmodyfikowanie innymi narzędziami.

TVTT_RandInit

Funkcja inicjuje generator liczb pseudolosowych

Deklaracja
void TVTT_RandInit(BYTE Init[16*4])

Parametry:

- Adres na blok danych inicjujących

Uwagi:
Funkcję tą należy wywołać przed jakąkolwiek inną funkcją operującą na generatorze liczb pseudolosowych.
Jako blok danych inicjujących należy użyć blok danych zwróconych przez funkcję TVTT_RandGetInit i zapisanych przy poprzednim zakończeniu programu. Jeżeli taki blok danych nie jest dostępny to należy użyć dowolnego bloku zawierającego możliwie losowe wartości.

Zasady komunikacji z TeleTokenem

Zaleca się, aby pierwszą operacją wykonaną na bibliotece przez program było sprawdzenie zgodności wersji API. Poprawne działanie programu możliwe jest tylko wtedy, gdy wersja API zwrócona przez bibliotekę jest dokładnie równa wersji API obsługiwanej przez program. Wersję API biblioteki można odczytać wywołując funkcję „TVTT_GetVersion”. Zwrócone przez tą funkcję informacje o wersji i podwersji biblioteki należy potraktować jako dane informacyjne i można je użyć przy wyświetlaniu informacji o programie oraz na potrzeby komunikatu błędu w przypadku stwierdzenia niezgodności API. Zaleca się wyświetlanie wersji biblioteki w postaci:

<wersja biblioteki>.<podwersja biblioteki>

przy czym jeśli podwersja jest mniejsza od dziesięciu to należy wyświetlić ją z jednym zerem poprzedzającym.

Przed rozpoczęciem komunikacji z TeleTokenem, należy otworzyć dostęp do niego korzystając z funkcji „TVTT_FindToken”. Funkcja ta zwraca „uchwyt na TeleToken”, który należy podawać podczas wywoływania wszystkich pozostałych funkcji operujących na TeleTokenie. W przypadku, gdy wiadomo, że program nie będzie już potrzebował się komunikować z danym TeleTokenem należy zwolnić zasoby zajęte na potrzeby komunikacji poprzez wywołanie funkcji „TVTT_FreeToken”. Również w przypadku wystąpienia poważnych błędów podczas komunikacji z TeleTokenem zalecane jest zwolnienie TeleTokenu i ponowne otwarcie dostępu do niego. Biblioteka dopuszcza możliwość równoległego otwarcia połączenia z różnymi TeleTokenami, przy czym każde takie otwarte połączenie otrzymuje własny „uchwyt”.

Wywołanie funkcji „TVTT_FindToken” powoduje otwarcie połączenia z TeleTokenem, nie powoduje jednak otwarcia kanału komunikacyjnego na potrzeby przesyłania danych i rozkazów do i z TeleTokenu. Przed rozpoczęciem komunikacji z TeleTokenem należy wywołać funkcję „TVTT_StartSession", która przygotowuje kanał komunikacji z TeleTokenem. Należy pamiętać, że do TeleTokenu można w danym momencie otworzyć tylko jeden kanał komunikacyjny, czyli użycie tej funkcji powoduje zablokowanie innym programom możliwości komunikacji z TeleTokenem. W przypadku, gdy kanał komunikacji jest otwarty przez inny program, funkcja „TVTT_StartSession" czeka aż inny program zamknie kanał i wtedy otwiera nowy kanał komunikacji, chyba, że wcześniej upłynie czas przeznaczony na wykonanie funkcji (w takim przypadku jest zwracany błąd „TVTTS_TIME_OUT"). Kanał komunikacyjny z TeleTokenem należy otwierać na możliwie krótki czas, aby umożliwić innym programom (w tym innym uruchomionym kopiom tego samego programu) komunikację z TeleTokenem.

Aby zamknąć kanał komunikacyjny z TeleTokenem należy wywołać funkcję „TVTT_EndSession".

Do wysyłania rozkazów oraz odbierania odpowiedzi z TeleTokenu służy funkcja „TVTT_Talk". Funkcja ta wymaga podania wektora inicjującego szyfrowanie, sam zwraca też taki wektor. Wektor przekazany do funkcji odnosi się do danych przesyłanych z PC do TeleTokenu, a wektor zwrócony przez funkcję odnosi się do danych przesyłanych z TeleTokenu do PC. Wartość wektora równa zero oznacza, że dane przesyłane są niezaszyfrowane. Jeżeli wektor jest różny od zera to znaczy, że przesyłane dane są zaszyfrowane a wartość wektora służy do wyliczenia (na podstawie klucza sesji) klucza tymczasowego do szyfrowania danych. Wartość początkowa wektora przekazywanego do funkcji powinna być możliwie nieprzewidywalna i niepowtarzalna. Kolejna wartość wektora przekazywana funkcji „TVTT_Talk" w obrębie sesji musi być większa od poprzedniej wartości wektora przekazanej do funkcji w obrębie tej samej sesji. Wektor inicjujący szyfrowanie musi przyjmować wartość zero (dane nie zaszyfrowane) dla rozpoczęcia sesji oraz rozkazu formatowania TeleTokenu, ponieważ nie ma jeszcze wynegocjowanego hasła sesji, więc nie ma możliwości

wygenerowania hasła tymczasowego do danych. W normalnych warunkach funkcja „TVTT_Talk" zwraca wektor równy zero tylko, jeśli wektor przekazany do funkcji był zerowy, a wektor niezerowy, jeśli wektor przekazany do funkcji był niezerowy. Jeśli funkcja zwróci wartość niezerową wektora w odpowiedzi na wektor zerowy to znaczy, że wystąpił poważny błąd w komunikacji lub nie została zachowana poprawna kolejność czynności podczas rozpoczynania sesji. Należy wówczas wysłać niezaszyfrowany rozkaz zakończenia sesji a następnie wywołać funkcję „TVTT_EndSession" ignorując kody błędów zwrócone przez obie operacje. Zalecane jest również odczekanie pewnego czasu (zależnie od możliwości kilku do kilkunastu sekund) przed rozpoczęciem następnej sesji.

Jeżeli w odpowiedzi na wektor niezerowy funkcja „TVTT_Talk" zwróciła wektor zerowy to znaczy, że TeleToken zakończył sesję i należy wówczas możliwie szybko wywołać funkcję „TVTT_EndSession". Aby poznać przyczynę zakończenia sesji należy sprawdzić zwróconą odpowiedź z TeleTokenu, przy czym odpowiedzi tej nie należy rozszyfrowywać.

Należy pamiętać, że wysłanie rozkazu otwarcia sesji do TeleTokenu nie musi spowodować bezwzględnego rozpoczęcia nowej sesji. Zawsze należy sprawdzać status zakończenia operacji zwracany przez TeleToken i zależnie od niego podejmować właściwe
działania. Szczególnie należy zwrócić uwagę na kod błędu „TVTTS_NO_OPEN”, który oznacza, że przekazany rozkaz (lub jego parametry lub dane) był w takim stopniu błędny, że jedynym bezpiecznym rozwiązaniem problemu było zakończenie sesji.

Pojedyncza „sesja” z TeleTokenem powinna rozpocząć się od wywołania funkcji „TVTT_StartSession”. Następnie należy wysłać rozkaz otwarcia sesji. Po otwarciu sesji należy w miarę potrzeb wysłać jeden lub więcej rozkazów do wykonania przez TeleToken. Po zakończeniu wykonywania operacji na TeleTokenie należy wysłać rozkaz zamknięcia sesji a następnie wywołać funkcję „TVTT_EndSession”.

Formatowanie TeleTokenu różni się od zwykłej sesji tym, że zamiast rozkazu otwarcia sesji należy wysłać rozkaz formatowania TeleTokenu. Następnie należy wywołać funkcję „TVTT_EndSession” (bez wysyłania rozkazu zamknięcia sesji) a następnie funkcję „TVTT_FreeToken”. Należy pamiętać, że po odebraniu poprawnego rozkazu formatowania, TeleToken nie odsyła żadnej odpowiedzi tylko od razu zaczyna się proces jego formatowania. Proces ten kończy się restartem TeleTokenu. Oznacza to, samo wysłanie rozkazu formatowania oraz wszystkie funkcje wywołane po nim mogą zwrócić błędy, które należy zignorować.

Wyszukiwanie TeleTokenu zawsze następuje po numerze identyfikacyjnym odbiorcy TeleTokenu, przy czym kolejność przeszukiwania portów USB jest niezdefiniowana. Jeżeli poda się ilość TeleTokenów do pominięcia równą zero to funkcja „TVTT_FindToken” zwróci uchwyt na pierwszy znaleziony TeleToken, którego identyfikator odbiorcy jest zgodny z identyfikatorem odbiorcy podanym przy wywołaniu funkcji. Jeżeli do systemu może być podłączone więcej niż jeden TeleToken o tym samym identyfikatorze odbiorcy, to pozostałe TeleTokeny można wyszukać wywołując w pętli funkcję „TVTT_FindToken” ze zwiększaną ilością TeleTokenów do pominięcia. Pętle należy przerwać po otrzymaniu kodu błędu:„TVTTS_TOKEN_NOT_FOUND”.

UWAGA! W specyficznych sytuacjach może się zdarzyć, że ten sam TeleToken zostanie odnaleziony więcej niż raz lub w ogóle nie zostanie odnaleziony. Taka sytuacja nie jest uważana za błąd, jednak zaleca się zabezpieczanie programów na wypadek wystąpienia takiego problemu.

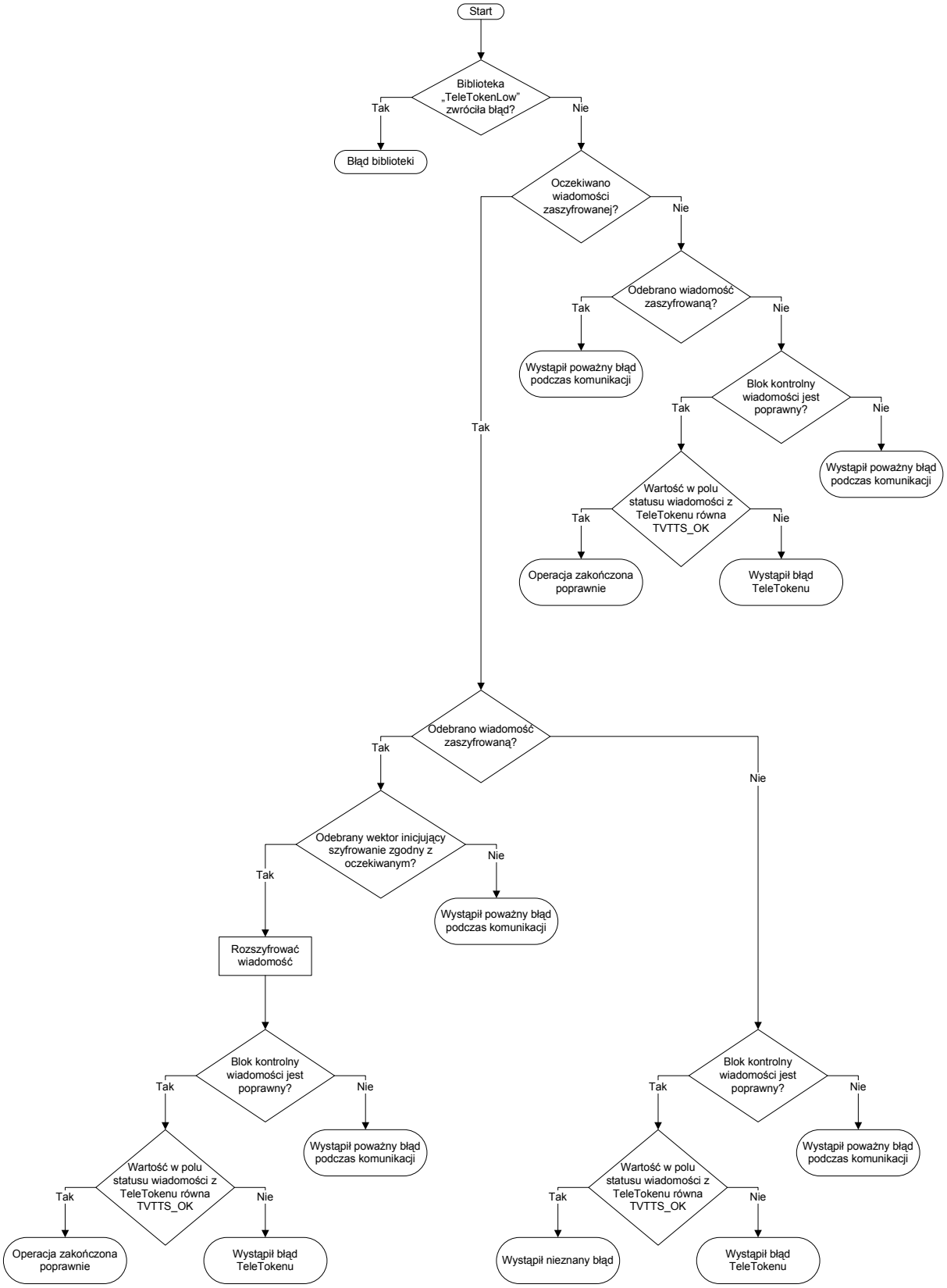
Maska dla ostatniego bajtu identyfikatora odbiorcy podawana funkcji „TVTT_FindToken” ustala, które bity z najmłodszego bajtu będą brane pod uwagę podczas wyszukiwania TeleTokenu. Jeśli bit w bajcie maski ma wartość jeden to odpowiadający mu bit w identyfikatorze odbiorcy podanym

funkcji „TVTT_FindToken” musi być identyczny z odpowiednim bitem identyfikatora zapisanego w TeleTokenie.

Przy wywołaniu funkcji TVTT_Talk potrzebne jest podanie wektora inicjującego szyfrowanie dla wiadomości wychodzącej. Należy zagwarantować, że w obrębie jednej sesji dla każdego wywołania funkcji TVTT_Talk wartość tego wektora będzie większa od użytej przy poprzednim wywołaniu TVTT_Talk.

Funkcja TVTT_Talk zwraca wektor inicjujący szyfrowanie dla wiadomości przychodzącej. O obrębie sesji należy zawsze sprawdzać czy wartość tego wektora zwrócona w wywołaniu funkcji TVTT_Talk jest o jeden (mod 2^24) większa od wartości zwróconej w poprzednim wywołaniu. Niepoprawna wartość wektora świadczy o poważnym błędzie podczas komunikacji lub o próbie ataku na aktualnie otwartą sesję.

Proponowany algorytm analizy błędów przedstawia rysunek:



W podanym algorytmie nie podano proponowanych sposobów reakcji na błędy, ponieważ obsługa błędów jest bardzo zależna od specyfiki programu oraz od przyjętych założeń odnośnie bezpieczeństwa. Decyzja o tym czy należy o danym typie błędu informować użytkownika czy nie również powinna być podjęta na podstawie analizy założeń odnośnie systemu, w którym TeleToken ma pracować.

Specyfikacja poszczególnych typów błędów:

- "Błąd TeleTokenu" - TeleToken wykrył błąd i zgłosił go w polu Statusu w odebranej wiadomości. Jeżeli wiadomość była odesłana jako zaszyfrowana to błąd nie był poważny i możliwe jest kontynuowanie sesji. Jeśli wiadomość była niezaszyfrowana to błąd był na tyle poważny, że nie możliwe jest kontynuowanie sesji, w związku, z czym, sesja została zakończona przez TeleToken. Błędy z tej kategorii mogą być spowodowane między innymi przez: podanie nieprawidłowych parametrów rozkazu, próbę wykonania niedozwolonej operacji, przekroczenie dopuszczalnego czasu trwania sesji, niezgodność haseł pomiędzy programem a TeleTokenem, uszkodzenie TeleTokenu
- "Błąd biblioteki" - biblioteka wykryła błędy podczas operacji na TeleTokenie i zwróciła jego kod do programu wywołującego.
- "Poważny błąd podczas komunikacji" - błędy, których klasyfikacja jest trudna i które zawsze oznaczają zakończenie sesji. Mogą powstać w skutek niezgodności haseł pomiędzy programem a TeleTokenem, problemów ze sprzętem (w tym z portami USB i okablowaniem), sterownikami lub programem. Błędy te mogą również wystąpić, jeśli w trakcie komunikacji z TeleTokenem nastąpi próba ataku na program, TeleToken, sterownik lub kanał komunikacyjny między programem a TeleTokenem.
- "Nieznany błąd" - błąd ten teoretycznie nie może wystąpić, ponieważ zgodnie z założeniami nie mogą być równocześnie spełnione wszystkie warunki konieczne do wystąpienia tego błędu. Wystąpienie tego błędu może być sygnałem poważnej awarii, błędu w systemie lub próby ataku na system.

Definicja stałych

Wszystkie zdefiniowane stałe mają przedrostek TVTT.

- Kody błędów zwracanych przez bibliotekę

Nazwa:	***Wartość:***	***Opis:***
TVTTS_OK	0	Operacja zakończona poprawnie
TVTTS_INVALID_HANDLE	0x0100	Niepoprawny uchwyt na TeleToken
TVTTS_NULL_PTR	0x0200	Adres na wymagany parametr jest wyzerowany
TVTTS_BAD_PARAM	0x0300	Podany parametr jest nie poprawny
TVTTS_TOO_MANY_OPEN_TOKENS	0x0400	Za dużo otwartych TeleTokenów
TVTTS_USB_DOES_NOT_EXIST	0x0500	Nie znaleziono portu USB
TVTTS_TOKEN_NOT_FOUND	0x0600	TeleToken nie znaleziony
TVTTS_TIME_OUT	0x0700	Przekroczenie czasu trwania operacji
TVTTS_SEQ_EXPIRED	0x0800	Sesja nie została otwarta lub wygasła
TVTTS_ALREADY_OPENED	0x0900	Sesja jest już otwarta
TVTTS_UNKNOWN_ERROR	0x1000	Nieznany typ błędu - zwracany, gdy nie da się jednoznacznie zakwalifikować błędu np.: operacja systemowa zwróciła nieoczekiwany kod błędu
TVTTS_DLL_INIT_FAILED	0xFF00	Błąd inicjalizacji biblioteki „TeleToken.dll

- Kody błędów zwracanych przez TeleToken w polu „Status”:

Nazwa:	***Wartość:***	***Opis:***
TVTTS_OK	0	Rozkaz wykonany poprawnie
TVTTS_NO_OPEN	0x11	Sesja nie została otwarta
TVTTS_EXPIRED	0x12	Sesja wygasła - Timeout
TVTTS_DAMAGED	0x13	Sesja nie otwarta - TeleToken uszkodzony
TVTTS_NOT_SUPPORTED	0x21	Rozkaz nie zaimplementowany
TVTTS_OUT_OF_RANGE	0x22	Adres, numer hasła lub numer licznika z poza zakresu
TVTTS_PROTECTION_NOT_SUPPORTED	0x23	Sektor, nie może zostać zabezpieczone przed zapisem, ponieważ nie posiada takiej możliwości
TVTTS_PASS_NOT_PERMITTED	0x31	Operacja niedopuszczalna dla hasła zaproponowanego podczas otwarcia sesji
TVTTS_NOT_PERMITTED	0x32	Operacja niedozwolona
TVTTS_ACCESS_EXPIRED	0x33	Przekroczona ilość operacji na sektorze, liczniku lub haśle

- Kody rozkazów TeleTokenu (pole Cmd):

Nazwa:	*Wartość:*	*Opis:*
TVTTC_OPEN	0x01	Otwarcie sesji komunikacji z TeleTokenem
TVTTC_CLOSE	0x02	Zamknięcie sesji komunikacji z TeleTokenem
TVTTC_FORMAT	0x0F	Sformatowanie TeleTokenu
TVTTC_NVMEM_READ	0x11	Odczyt z pamięci nieulotnej TeleTokenu
TVTTC_NVMEM_WRITE	0x12	Zapis do pamięci nieulotnej TeleTokenu
TVTTC_NVMEM_WRITE_PROTECT	0x13	Zapis do pamięci nieulotnej TeleTokenu i zablokowanie możliwości zapisu do pamięci
TVTTC_NVMEM_PROTECT	0x14	Zapis i zablokowanie możliwości zapisu do sektora
TVTTC_VMEM_READ	0x91	Odczyt z pamięci ulotnej TeleTokenu
TVTTC_NVMEM_WRITE	0x92	Zapis do pamięci ulotnej TeleTokenu
TVTTC_ENCRYPT	0x21	Zaszyfrowanie bloku danych kluczem zapisanym w tablicy haseł TeleTokenu
TVTTC_DECRYPT	0x22	Zaszyfrowanie bloku danych kluczem zapisanym w tablicy haseł TeleTokenu
TVTTC_CNT_READ	0x31	Odczyt zawartości z licznika nieulotnego
TVTTC_CNT_INC_READ	0x32	Zwiększenie o jeden a następnie odczyt zawartości z licznika nieulotnego
TVTTC_CNT_DEC_READ	0x33	Zmniejszenie o jeden a następnie odczyt zawartości z licznika nieulotnego
TVTTC_CNT_WRITE	0x34	Zapis do licznika nieulotnego
TVTTC_CNT_INC_ENCRYPT_READ	0x35	Zwiększenie o jeden a następnie odczyt zawartości z licznika nieulotnego. Odczytana wartość przed przesłaniem do PC jest szyfrowana hasłem o podanym numerze.
TVTTC_CNT_WRITE_CONFIG	0x36	Wpisanie bitów konfiguracji licznika
TVTTC_CNT_SET_CONFIG	0x37	Ustawienie bitów konfiguracji licznika
TVTTC_CNT_CLR_CONFIG	0x38	Skasowanie bitów konfiguracji licznika
TVTTC_PASS_WRITE	0x41	Wpisanie hasła do tablicy haseł TeleTokenu
TVTTC_PASS_WRITE_CONFIG	0x42	Wpisanie bitów konfiguracji hasła

Nazwa:	***Wartość:***	***Opis:***
TVTTC_PASS_SET_CONFIG	0x43	Ustawienie bitów konfiguracji hasła
TVTTC_PASS_CLR_CONFIG	0x44	Skasowanie bitów konfiguracji hasła

- Bity konfiguracji licznika:

Nazwa:	*Bajt:*	*Wartość:*	*Opis:*
TVTTCC0_ WRITE	0	0x01	Licznik może być zapisywany
TVTTCC0_ READ	0	0x02	Licznik może być odczytany bez modyfikacji
TVTTCC0_ INC_READ	0	0x04	Licznik może być jednym rozkazem zwiększony i odczytany
TVTTCC0_ DEC_READ	0	0x08	Licznik może być jednym rozkazem zmniejszony i odczytany
TVTTCC0_ INC_ENCRYPT_READ	0	0x10	Licznik może być jednym rozkazem zwiększony i odczytany a odczytana wartość ma zostać przed przesłaniem zaszyfrowana hasłem o podanym numerze
TVTTCC0_ STOP_IN_ZERO	0	0x20	Licznik ma być zatrzymany po dojściu do zera
TVTTCC0_ NVMEM_READ	0	0x40	Licznik jest powiązany z odczytem pamięci
TVTTCC0_ NVMEM_WRITE	0	0x80	Licznik jest powiązany z zapisem pamięci
TVTTCC1_ PASS	1	0x01	Licznik jest powiązany z hasłem w tablicy
TVTTCC1_CONFIG_ENABLED	1	0x80	Odblokowanie możliwości modyfikowania bitów konfiguracji licznika (UWAGA! Ten bit można tylko skasować!)

- Bity konfiguracji hasła:

Nazwa:	*Bajt:*	*Wartość:*	*Opis:*
TVTTPC0_TOKEN	0	0x01	hasło może być proponowane przez TeleToken do generowania klucza sesji
TVTTPC0_PASS_WRITE	0	0x02	hasło może być proponowane przez PC do generowania klucza sesji wpisania hasła
TVTTPC0_PASS_CONFIG	0	0x04	hasło może być proponowane przez PC do generowania klucza sesji modyfikacji bitów konfiguracji hasła
TVTTPC0_MEM_READ	0	0x08	hasło może być proponowane przez PC do generowania klucza sesji odczytu pamięci
TVTTPC0_MEM_WRITE	0	0x10	hasło może być proponowane przez PC do generowania klucza sesji zapisu pamięci
TVTTPC0_ENCRYPT	0	0x20	hasło może być proponowane przez PC do generowania klucza sesji szyfrowania
TVTTPC0_DECRYPT	0	0x40	hasło może być proponowane przez PC do generowania klucza sesji deszyfrowania
TVTTPC1_CNT_READ	1	0x01	hasło może być proponowane przez PC do generowania klucza sesji odczytu licznika
TVTTPC1_CNT_WRITE	1	0x02	hasło może być proponowane przez PC do generowania klucza sesji zapisu licznika
TVTTPC1_CNT_CONFIG	1	0x04	hasło może być proponowane przez PC do generowania klucza sesji modyfikacji bitów konfiguracji licznika
TVTTPC1_ENCRYPT_PASS	1	0x08	hasło może być używane do szyfrowania
TVTTPC1_DECRYPT_PASS	1	0x10	hasło może być używane do rozszyfrowywania
TVTTPC1_CNT_ENCRYPT_PASS	1	0x20	hasło może być używane do szyfrowania wartości odczytanej z licznika (praca w trybie generacji haseł uwierzytelniających)
TVTTPC1_WRITE_ENABLED	1	0x40	odblokowanie możliwości nadpisania hasła (UWAGA! Ten bit można tylko skasować!)
TVTTPC1_CONFIG_ENABLED	1	0x80	odblokowanie możliwości modyfikowania bitów konfiguracji hasła (UWAGA! Ten bit można tylko skasować!)

Interpretacja wybranych kodów statusu

TVTTS_BAD_PARAM

Może wystąpić, gdy:

- Podana wartość jednostek czasu przeznaczonego na wykonanie funkcji jest z poza zakresu.
- Podana ilość TeleTokenów do pominięcia jest z poza zakresu.
- Podano sprzeczne parametry np.: przy wywołaniu „TVTT_FindToken” podano rozmiar tablicy do wypełnienia a nie podano adresu na tablicę.

TVTTS_USB_DOES_NOT_EXIST

Błąd ten oznacza, że podsystem obsługi USB zwrócił poważny błąd lub nie udało się odnaleźć podsystemu USB. Prawdopodobnie w komputerze nie ma zainstalowanych portów USB, zainstalowany system operacyjny nie obsługuje portów USB (Windows 95, Windows NT w wersji 4.0 i niższej etc.) lub sterowniki USB nie działają poprawnie.

TVTTS_SEQ_EXPIRED

Może wystąpić, gdy:

- Wywołano funkcję „TVTT_Talk”, mimo, że wcześniej nie rozpoczęto komunikacji z TeleTokenem poprzez wywołanie funkcji „TVTT_StartSession” (lub po „TVTT_StartSession” a przed „TVTT_Talk” wywołano „TVTT_EndSession”).
- Wystąpił poważny problem podczas komunikacji z TeleTokem i nastąpiła utrata synchronizacji pomiędzy TeleTokenem a sterownikiem. Należy natychmiast wywołać funkcję „TVTT_EndSession”, po czym można spróbować ponownie rozpocząć komunikację z TeleTokenem.

TVTTS_ALREADY_OPENED

Błąd zgłaszany, gdy wywołano funkcję „TVTT_StartSession” a komunikacja z TeleTokenem jest już otwarta przez wcześniejsze wywołanie tej funkcji. Zawsze pomiędzy każdymi dwoma wywołaniami funkcji „TVTT_StartSession” musi znaleźć się wywołanie „TVTT_EndSession”. Nie jest błędem wywołanie funkcji „TVTT_EndSession” bez wcześniejszego wywołania „TVTT_StartSession” lub, gdy wcześniejsze wywołanie „TVTT_StartSession” zakończyło się niepowodzeniem. (Nie da się ponownie otworzyć już otwartej sesji, ale dopuszcza się ponowne zamkniecie już zamkniętej sesji)

TVTTS_UNKNOWN_ERROR

Błąd zgłaszany, gdy wystąpiło jakieś nieprzewidziane zdarzenie nie obsługiwane przez sterownik TeleTokenu lub, gdy któryś z wewnętrznych testów nadmiarowych wykazał niepoprawne funkcjonowanie sterownika.

TVTTS_DLL_INIT_FAILED

W przypadku biblioteki dla języka „C” i „C++” załadowanie sterownika TeleTokenu (zawartego w bibliotece „TeleToken.dll”) następuje dopiero podczas pierwszego wywołania którejkolwiek z funkcji biblioteki. Jeżeli próba załadowania sterownika zakończy się niepowodzeniem to funkcja zwróci błąd a funkcja systemowa „GetLastError” zwróci poprawny kod opisujący przyczynę wystąpienia błędu.

Stan TeleTokenu po sformatowaniu

Ponieważ formatowanie ma na celu odzyskanie kontroli nad TeleTokenem, więc samo wyczyszczenie pamięci TeleTokenu nie jest wystarczające. Konieczne jest dokonanie takich wstępnych wpisów do pamięci, aby udostępnić minimalną funkcjonalność niezbędną do odzyskania kontroli nad TeleTokenem.

W ramach formatowania TeleTokenu zostaje wykonane:

- Wyzerowanie wszystkich bitów konfiguracji haseł
- Wyzerowanie wszystkich bitów konfiguracji liczników
- Wyzerowanie wszystkich sektorów pamięci użytkownika
- Wyzerowanie wszystkich liczników
- Wyzerowanie wszystkich haseł w tablicy
- Odblokowanie zapisu do wszystkich sektorów pamięci
- Odblokowanie możliwości modyfikowania bitów konfiguracji wszystkim licznikom
- Odblokowanie możliwości modyfikowania wszystkich haseł oraz bitów konfiguracji wszystkich haseł
- Ustawienie pierwszemu hasłu w tablicy bitu zezwolenia na zaproponowanie go przez PC do generowania klucza sesji modyfikacji bitów konfiguracji hasła
- Ustawienie drugiemu hasłu w tablicy bitu zezwolenia na zaproponowanie go przez TeleToken do generowania klucza sesji

UWAGA! Klucz sesji jest liczony wyłącznie przy otwarciu sesji. Oznacza to, że zmiany haseł używanych do szyfrowania aktualnej sesji odnoszą skutek dopiero po otwarciu nowej sesji!