BIULETYN INSTYTUTU AUTOMATYKI I ROBOTYKI
NR 17, 2002

Propozycja „utwardzenia" systemu operacyjnego AS/400

Piotr NOWAK

Wydział Cybernetyki WAT, ul. Kaliskiego 2, 00-908 Warszawa

STRESZCZENIE: W artykule przedstawiono ogólne informacje na temat bezpieczeństwa systemu AS/400. Omówiono poziomy bezpieczeństwa, rodzaje profili użytkownika wraz z uprawnieniami, zaproponowano trzy modele autoryzacji oraz wartości zmiennych systemowych bezpośrednio odpowiedzialnych za bezpieczeństwo.

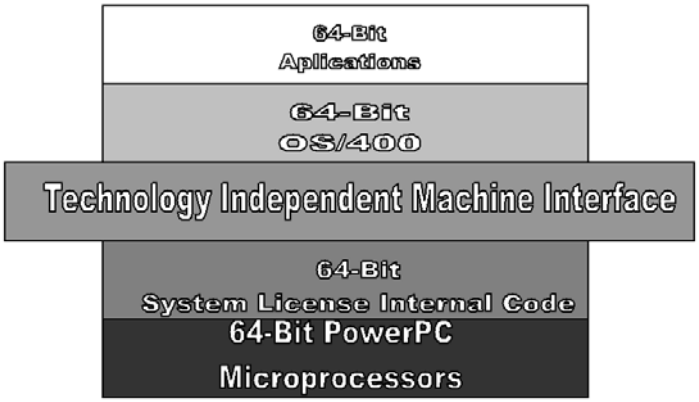
1. Wstęp

AS/400[1] (Application System/400) jest mało popularnym systemem komputerowym wśród informatyków, a wykorzystywanym głównie w instytucjach finansowych i wielkich korporacjach o zasięgu międzynarodowym. Jest uwazany jako jeden z najbezpieczniejszych na świecie. Jest systemem klasy C2 według kryteriów Departamentu Obrony Narodowej USA. Systemem operacyjnym jest OS/400[2].

Cechą, która wyróżnia AS/400 spośród innych systemów komputerowych, jest jego architektura. Oddziela ona system operacyjny i aplikacje od technologii sprzętowej interfejsem TIMI (Technology Independent Machine Interface). Niezależność ta oznacza możliwość natychmiastowego dostępu do procesorów najnowszych generacji. Po przeniesieniu na nowy komputer wszystkie dotychczas istniejące aplikacje działają natychmiast,

[1] Obecnie w nomenklaturze IBM jest to komputer serii I – iSeries.
[2] Od niedawna IBM wprowadził Linux na AS/400.

w pełni wykorzystując nową technologię procesora. Od 1995 roku AS/400 wykorzystuje 64-bitowy RISC PowerPC.



Rys. 1. Schemat architektury AS/400

AS/400 jest produktem powstałym z połączenia najlepszych cech wcześniejszych produktów firmy IBM znanych pod nazwami System/36 i System/38[3]. W S/36, administrator mógł wybrać ochronę zasobów lub ochronę tylko poprzez hasło. Natomiast w S/38 hasło oraz ochrona zasobów są wymagane. OS/400 łączy te dwie rzeczy, a mianowicie pozwala wybrać odpowiedni poziom bezpieczeństwa.

W dalszej części artykułu zostaną przedstawione następujące elementy: poziomy bezpieczeństwa, profile użytkownika, proponowane modele autoryzacji oraz zmienne systemowe i ich proponowane wartości.

Po każdym rozdziale występuje część „SPRAWDZENIE", w której zaleca się Czytelnikowi sprawdzenie ustawień używanego systemu z proponowanymi przez autora poziomami.

2. Poziom bezpieczeństwa

Przy użyciu zmiennej systemowej QSECURITY ustawiamy właściwy dla naszej organizacji poziom bezpieczeństwa. Dostępne są poziomy oznaczone wartościami: 20, 30, 40, i 50. W starszych wersjach systemu niż V4R3 dostępny był jeszcze poziom bezpieczeństwa 10, który obecnie już nie jest implementowany w systemie.

[3] Zwane dalej S/36 i S/38.

Poziom 10 potocznie zwany był poziomem 0 ponieważ jedynym zabezpieczeniem była ochrona fizyczna dostępu do komputera. Jeśli użytkownik miał dostęp do konsoli z ekranem logowania, mógł wpisać nazwę użytkownika i zalogować się (hasło nie było wymagane). Gdy podana nazwa użytkownika nie istniała, system automatycznie tworzył profil nowego użytkownika i pozwalał na pracę. Nowo tworzony użytkownik dostawał uprawnienie *ALLOBJ, które pozwala na kasowanie i modyfikacje wszystkich danych w systemie, co jest BARDZO NIEBEZPIECZNE !

Poziom 20 dostarcza ochronę przez hasło. Aby uzyskać dostęp do systemu działającym na poziomie bezpieczeństwa 20, użytkownik musi znać nazwę profilu użytkownika stworzonego w systemie oraz jego hasło dostępu. Dzięki tym zabezpieczeniom w minimalnym stopniu udaje się ograniczyć nieautoryzowany dostęp do systemu.
Od tego poziomu możemy nadawać użytkownikom ograniczone prawa dostępu do danych (Uprawnienia specjalne), oraz możemy ustawić zmienną LMTCPB na *YES, co uniemożliwi wydawanie poleceń z linii komend użytkownikowi. Będzie on mógł korzystać z gotowych, przygotowanych opcji w menu. Uprawnienia specjalne przedstawia tab. 1.

Tab. 1. Uprawnienia specjalne dla Poziomu 20[4]

Typ użytkownika	**Uprawnienia specjalne**				
*SECOFR	*ALLOBJ	*SECADM	*SAVSYS	*JOBCTL	*SERVICE
	*SPLCTL	*AUDIT	*IOSYSCFG		
*SECADM	*ALLOBJ	*SECADM	*SAVSYS		
*PGMR	*ALLOBJ		*SAVSYS		
*SYSOPR	*ALLOBJ		*SAVSYS	*JOBCTL	
*USER	*ALLOBJ		*SAVSYS		

Jak zauważamy, każdy nowotworzony użytkownik dostaje domyślnie uprawnienie *ALLOBJ. Administrator, jeśli chce ustawić poziom 20, musi usunąć te uprawnienie i pozostawić je tylko użytkownikom, którym jest to niezbędne, inaczej w systemie może dochodzić do nieautoryzowanego dostępu do danych.

Poziom 30 przy tworzeniu nowego profilu nie ustawia *ALLOBJ jako domyślne uprawnienie. Daje to ochronę danych, ponieważ dostęp do zasobów jest przydzielany przez administratora, a nie jak w poziomie 20 każdy użytkownik domyślnie miał dostęp do wszystkich danych w systemie.

[4] Szczegółowo uprawnienia specjalne omówiono przy profilu użytkownika.

Tab. 2. Uprawnienia specjalne dla Poziomu 30, 40, 50

Typ użytkownika	Uprawnienia specjalne				
*SECOFR	*ALLOBJ	*SECADM	*SAVSYS	*JOBCTL	*SERVICE
	*SPLCTL	*AUDIT	*IOSYSCFG		
*SECADM		*SECADM			
*PGMR					
*SYSOPR			*SAVSYS	*JOBCTL	
*USER					

Jak pokazano tab. 2, w poziomie 30 (również 40 i 50) uprawnienia specjalne są bardzo restrykcyjne i tylko użytkownik typu Security Oficer posiada *ALLOBJ nadawane domyślnie przy tworzeniu profilu. Użytkownicy nie posiadają domyślnie praw do tworzenia, modyfikowania czy kasowania obiektów w systemie. Te uprawnienia muszą być nadane świadomie przez administratora.

Podczas przechodzenia z poziomu 10 lub poziomu 20 do poziomu 30, OS/400 usunie specjalne uprawnienia użytkownikom, którzy w poziomie 30 nie posiadają ich domyślnie. Należy więc tę operację wcześniej zaplanować by przywrócić użytkownikom uprawnienia niezbędne do prawidłowej pracy.

Większość systemów AS/400, z którymi autor zetknął się, pracowało na poziomie bezpieczeństwa 30, który powszechnie uznaje się za zbyt niski.

Poziom 40 pozwala na uszczelnienie dziur w systemie bezpieczeństwa, które zostały odziedziczone z S/38. Te luki pozwalały doświadczonym programistom na nieautoryzowany dostęp do obiektów oraz danych (tzw. bakcdoors). Poziom 40 zapewnia integralność systemu operacyjnego, między innymi poprzez zabranianie odtwarzania uszkodzonych lub modyfikowanych programów, ograniczanie listy dostępnych instrukcji MI (Machine Interface). Uważa się, że jest to minimalny poziom, na jakim powinien pracować system działający w korporacjach, do którego ma dostęp kilkudziesięciu i więcej użytkowników.

Poziom 50 dostarcza dodatkowych funkcji zabezpieczających system, wymaganych przez Departament Obrony USA dla systemów klasy C2. Między innymi system sprawdza poprawność podawanych parametrów przed uruchomieniem programu.

Organizacje, które wymagają najwyższych zabezpieczeń, powinny zwiększyć bezpieczeństwo systemu AS/400 do poziomu 50. Po przejściu na poziom 50 może okazać się, że wiele aplikacji przestanie działać poprawnie albo w ogóle nie da się uruchomić. Świadczyć to będzie o ich amatorskiej budowie. Bardzo często spotykamy się z sytuacją, gdy dostawca

oprogramowania podczas instalacji i pokazu swojego produktu ustawia poziom bezpieczeństwa na 30. Administrator nie powinien na to pozwolić!

SPRAWDZENIE:

- Sprawdź, czy zmienna systemowa QSECURITY jest ustawiona na 40 lub 50.
- Sprawdź, czy pozostałe zmienne systemowe są ustawione zgodnie z zaleceniem z tabeli 3.

3. Zmienne systemowe

Ustawienie odpowiedniego poziomu bezpieczeństwa poprzez zmienną QSECURITY jest tylko fundamentem do dalszej budowy bezpiecznego systemu. Producent AS/400 wraz z systemem operacyjnym dostarczył dodatkowych zmiennych systemowych, dzięki którym można zabezpieczyć system. W niniejszym artykule nie będzie szczegółowego omówienia każdej z nich, ponieważ celem artykułu jest tylko zasygnalizowanie ich istnienia i pobudzenie do własnej, głębszej analizy.

Do pracy ze zmiennymi systemowymi należy posłużyć się komendami: WRKSYSVAL[5], CHGSYSVAL[6], DSPSYSVAL[7]. Do zmiennych systemowych odpowiadających bezpośrednio za bezpieczeństwo systemu zalicza się[8]:

- QALWOBJRST — określa, czy obiekty chronione mogą być odtwarzane.
- QALWUSRDMN — podaje, które biblioteki w systemie mogą zawierać obiekty typu *USRSPC, *USRIDX, *USRQ.
- QAUDCTL — określa, czy zapisywanie zdarzeń jest aktywne.
- QAUDENDACN — definiuje działanie systemu, gdy zapisy kontroli nie mogą być przesłane do dziennika z powodu błędów.

[5] WRKSYSVAL (work with system value) — komenda ta umożliwia prace ze zmiennymi systemowymi.
[6] CHGSYSVAL (change system value) — komenda ta umożliwia zmianę wartości zmiennej systemowej.
[7] DSPSYSVAL (display system value) — komenda ta służy do wyświetlania bieżącej wartości zmiennych
systemowych.
[8] Są to zmienne w wersji systemu 4 i starszych.

- QAUDFRCLVL — określa liczbę zapisów w dzienniku kontroli ochrony zanim dane zostaną przesłane do pamięci.
- QAUDLVL — określa, jakie działania są kontrolowane w systemie.
- QCRTAUT — tworzy domyślne uprawnienia publiczne dla komend tworzenia (CRTxxx), które mogą być dostępne w całym systemie.
- QCRTOBJAUD — określa domyślną wartość kontroli, używaną gdy w bibliotece tworzone są obiekty.
- QDSPSGNINF — włącza lub wyłącza wyświetlanie ekranu informacji o zalogowaniu się.
- QINACTITV — podaje w minutach dozwolony czas nieaktywności zadań.
- QINACTMSGQ — określa kolejkę komunikatów o nieaktywności. Określa ona działanie podejmowane przez system, gdy zadanie interaktywne pozostaje nieaktywne przez określony czas.
- QLMTDEVSSN — umożliwia ograniczenie równoczesnych sesji urządzeń.
- QLMTSECOFR — umożliwia ograniczenie dostępu do stacji roboczych przez użytkowników z uprawnieniami specjalnymi.
- QMAXSGNACN — określa działanie systemu po osiągnięciu maksymalnej liczby prób zalogowania się.
- QMAXSIGN —określa maksymalną dopuszczalna liczba nieudanych prób zalogowania się.
- QPWDEXPITV — określa liczbę dni ważności hasła.
- QPWDLMTAJC — ogranicza użycie sąsiadujących ze sobą cyfr w haśle.
- QPWDLMTCHR — ogranicza użycie pewnych znaków w haśle.
- QPWDLMTREP — ogranicza użycie powtarzających się znaków w haśle.
- QPWDMAXLEN — określa maksymalną liczbę znaków w haśle.
- QPWDMINLEN — określa minimalną liczbę znaków w haśle.
- QPWDPOSDIF — określa położenie znaków w nowym haśle. Pozwala to uniknąć występowania tych samych znaków w tym samym miejscu, co w poprzednim haśle.
- QPWDRQDDGT — określa, czy w nowym haśle musi występować cyfra.
- QPWDRQDDIF — określa, czy hasło musi różnić się od poprzednich.

- QPWDVLDPGM — podaje nazwę programu sprawdzania hasła dostarczonego podczas instalacji.
- QRETSVRSEC — wartość systemowa sterująca przechowywaniem danych ochrony serwera.
- QRMTSIGN — zarządza zdalnym wpisywaniem się.
- QSECURITY — określa systemowy poziom bezpieczeństwa.
- QUSEADPAUT — użycie uprawnień przejmowanych.

W tab. 3 autor przedstawił propozycję wartości zmiennych systemowych odpowiedzialnych za bezpieczeństwo.

4. Profil użytkownika i jego uprawnienia

Bardzo ważnym elementem bezpieczeństwa systemu AS/400 jest odpowiednie skonfigurowanie profilu użytkownika. Profil użytkownika należy do obiektów systemu najbardziej uniwersalnych i o największej mocy. Zawiera on hasło użytkownika i określa menu, jakie widzi użytkownik po wpisaniu się do systemu. Definiuje także, jakie działania użytkownik może, a jakich nie może wykonywać w systemie. Określa, w jaki sposób dany użytkownik widzi system. Konfiguracja profilu nowotworzonego użytkownika jest prosta, lecz trzeba się zastanowić przy nadawaniu typu użytkownika i uprawnień specjalnych. Pozostałe parametry można pozostawić jako domyślne, z wyjątkiem hasła. Zainteresowanemu Czytelnikowi poleca się literaturę [1], [2].

Do pracy z profilami użytkowników należy posłużyć się komendami: CRTUSRPRF[9], CHGUSRPRF[10], DLTUSRPRF[11], DSPUSRPRF[12].

Parametr określający typ użytkownika (USRCLS) oraz parametr określający uprawnienia specjalne (SPCAUT) są ze sobą powiązane. Jako typ użytkownika (USRCLS) można wybrać jeden z pięciu dostępnych:

[9] CRTUSRPRF (create user profile) — komenda umożliwia stworzenie nowego profilu użytkownika.
[10] CHGUSRPRF (change user profile) — komenda umożliwia zmianę ustawień w profilu użytkownika.
[11] DLTUSRPRF (delete user profile) — komenda umożliwia kasowanie profilu użytkownia.
[12] DSPUSRPRF (display user profile) — komenda umożliwia wyświetlenie bieżących wartości parametrów profilu użytkownika.

1) *SECOFR — security officer
2) *SECADM— security administrator
3) *PGMR— programmer
4) *SYSOPR— system operator
5) *USER— user

Tab. 3. Propozycja szablonu sprawdzania wartości zmiennych systemowych

ZMIENNA SYSTEMOWA	WARTOŚĆ ZALECANA	AKTUALNA WARTOŚĆ	ZGODNOŚĆ Z ZALECENIEM (TAK / NIE)
QALWOBJRST	*NONE		
QALWUSRDMN	*ALL		
QAUDCTL	*AUDLVL		
QAUDENDACN	*NOTIFY		
QAUDFRCLVL	*SYS		
QAUDLVL	*AUTFAIL,		
	*DELETE,		
	*SAVRST,		
	*SECURITY		
QCRTAUT	*CHANGE		
QCRTOBJAUD	*NONE		
QDSPSGNINF	‘0’ = No		
QINACTITV	30		
QINACTMSGQ	*DSCJOB		
QLMTDEVSSN	‘1’ = Yes		
QLMTSECOFR	‘1’ = Yes		
QMAXSGNACN	3		
QMAXSIGN	3		
QPWDEXPITV	60		
QPWDLMTAJC	‘1’ = Yes		
QPWDLMTCHR	*NONE		
QPWDLMTREP	2		
QPWDMAXLEN	10		
QPWDMINLEN	7		
QPWDPOSDIF	‘0’ = No		
QPWDRQDDGT	‘1’ = Yes		
QPWDRQDDIF	6		
QPWDVLDPGM	*NONE		
QRETSVRSEC	‘0’ = No		
QRMTSIGN	*SAMEPRF		
QSECURITY	40 lub 50		
QUSEADPAUT	USEADOPTED		

Dzięki temu można poklasyfikować użytkowników według ich typów. Tabele 1 i 2 przedstawiają uprawnienia odpowiednich typów użytkowników. Natomiast, gdy jest potrzeba stworzyć profil użytkownika o nietypowych uprawnieniach należy posłużyć się uprawnieniami specjalnymi :

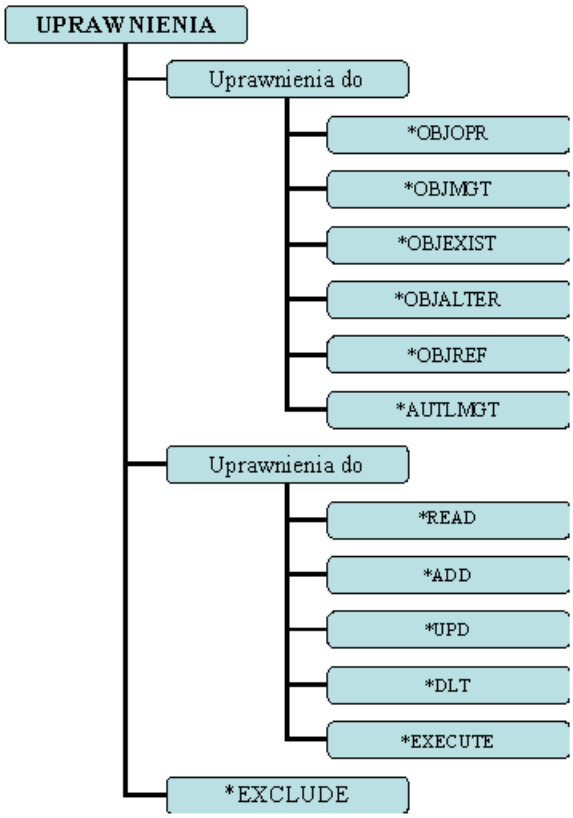
1) *USRCLS — uprawnienia są przypisywane według typu użytkownika.
2) *NONE— brak dodatkowych uprawnień.
3) *ALLOBJ— dostęp do wszystkich obiektów w systemie.
4) *AUDIT— użytkownik posiadający to uprawnienie, ma możliwość konfiguracji zapisu zdarzeń w systemie.
5) *IOSYSCFG— uprawnienia do konfiguracji systemu i zmiany atrybutów
6) *JOBCTL— uprawnienia do zmiany, wyświetlania, wstrzymywania, wznawiania, anulowania zadań z kolejki.
7) *SAVSYS — możliwość składowania i odtwarzania obiektów w systemie.
8) *SECADM — uprawnienie do zakładania kont użytkowników[13].
9) *SERVICE — uprawnienie do korzystania z System Service Tools.
10) *SPLCTL — uprawnienie do usuwania, wyświetlania, wstrzymywania, wznawiania wydruków w kolejce.

Nadanie uprawnień użytkownikowi musi być przemyślane i uzasadnione, ponieważ niezdyscyplinowany użytkownik może być potężnym zagrożeniem dla systemu. Jeśli chodzi o uprawnienie *ALLOBJ to powinien posiadać je tylko użytkownik typu *SECOFR.

Na rys. 2 przedstawiono dwa rodzaje uprawnień: uprawnienia do obiektu (object authorities) i uprawnienia do danych (data authorities).

Uprawnienia do obiektu zezwalają na działania takie jak: zmiana nazwy obiektu, zamiana zawartości obiektu, zapisywanie i kasowanie obiektu.

[13] Użytkownik z tym uprawnieniem może zakładać konta innym użytkownikom, którzy nie wymagają konta
typu *SECOFR oraz uprawnienia *ALLOBJ. Takie konto może założyć tylko użytkownik typu *SECOFR.



Rys. 2. Schemat przedstawiający dwa rodzaje uprawnień

Natomiast uprawnienia do danych mają zastosowanie podczas dodawania, uaktualniania, czytania i kasowania zawartości obiektu np. rekordu. Uprawnienia do obiektu podzielone są na trzy kategorie. Do pierwszej zaliczono uprawnienia, które mają zastosowanie do wszystkich obiektów w systemie. Są to:

1) *OBJOPR — (Object Operation) uprawnienie to zezwala na używanie obiektu, czyli obejrzenie opisu obiektu. Jeśli obiekt zawiera dane lub wykonywalny kod programu, użytkownik ma dostęp do opisu obiektu ale nie ma dostępu do danych oraz nie może uruchomić tego kodu.
2) *OBJMGT — (Object Management) uprawnienie to zezwala użytkownikowi na przesunięcie i zmianę nazwy obiektu, zmiany atrybutów obiektu oraz nadawanie uprawnień do tego obiektu innym użytkownikom.

3) *OBJEXIST — (Object Existence) uprawnienie to zezwala użytkownikowi na kasowanie, składowanie i odtwarzanie obiektu oraz przekazywanie prawa własności do tego obiektu.Aby zmniejszyć ryzyko przed niewłaściwym i nieprzemyślanym działaniem użytkowników należy zostawić prawa *OBJMGT i *OBJEXIST tylko właścicielom obiektów.

Do drugiej kategorii zaliczono uprawnienia, które mają zastosowanie w plikach baz danych. Są to:

1) *OBJALTER — (Object Alter) uprawnienie to zezwala użytkownikowi na dodawanie, czyszczenie, przestawianie i uruchamianie membrów, oraz zmianę atrybutów w plikach baz danych.
2) *OBJREF— (Object Reference) uprawnienie to zezwala użytkownikowi na zmianę powiązań między plikami w bazie danych. To uprawnienie powinien posiadać administrator bazy danych lub odpowiedzialny programista.

Trzecią grupę uprawnień do obiektu stanowi uprawnienie *AUTLMGT. Uprawnienie to zezwala na dodawanie i usuwanie uprawnień użytkowników z listy uprawnień danego obiektu.

Uprawnienia do danych:

1) *READ — zezwala użytkownikowi na czytanie i wyszukiwanie danych.
2) *ADD — zezwala użytkownikowi wstawiać nowe rekordy.
3) *UPD — zezwala użytkownikowi mod3pI(kowstawia)]TJ/TT1 0 617[1.65 0 TD0 Tc0 Tw0 0fcTj/TT

Przykład:
Załóżmy, że biblioteka NOWY zawiera plik (physical file) ORNETA. Użytkownik, który chce mieć dostęp do pliku ORNETA musi mieć uprawnienia *OBJOPR i *READ lub *EXECUTE do biblioteki NOWY. Aby skasować plik ORNETA użytkownik potrzebuje prawa dostępu *OBJOPR, *READ i *EXECUTE do biblioteki NOWY oraz *OBJEXIST do pliku ORNETA. Aby skasować bibliotekę użytkownik potrzebuje uprawnienia *OBJEXIST do biblioteki.

SPRAWDZENIE:

- Sprawdź profile posiadające dodatkowe uprawnienia.
- Sprawdź profile grupowe.
- Sprawdź profile użytkowników posiadające taką samą nazwę jak hasło.
- Sprawdź prawa własności do obiektów.
- Sprawdź uprawnienia do krytycznych obiektów w systemie.
- Sprawdź funkcje, które mogą wykonywać użytkownicy.

5. Konie trojańskie

Konie trojańskie są najbardziej popularnym narzędziem napastnika próbującego zwiększyć swe uprawnienia w systemie AS/400. Atakujący, który chce pomyślnie ulokować swojego konia trojańskiego, szuka okazji by zastąpić swym trojanem normalny program np. zmieniając nazwę na często wykonywaną komendę w systemie lub nadać nazwę i opis swojemu programowi tak, by zachęcał do uruchomienia i nie wzbudzał żadnych podejrzeń. W AS/400 możemy wykorzystać do tego celu bibliotekę QGPL, która domyślnie zezwala użytkownikom na zmiany swej zawartości, ponieważ ma atrybut *CHANGE. Administrator powinien zmienić to ustawienie lub często wykonywać kontrolę zawartości tej biblioteki.

SPRAWDZENIE:

- Dokładnie sprawdź biblioteki poprzedzające bibliotekę QSYS na liście bibliotek (najlepiej by biblioteka QSYS była pierwszą na liście bibliotek).
- Szczegółowo i skrupulatnie planuj listę bibliotek dla zadań – część systemową, część produkcyjną i część użytkownika.

- Zabezpiecz opisy podsystemów tak, by zabronić dokonywania zmian przez osoby nieupoważnione.
- Przełącz system na poziom bezpieczeństwa 40 lub 50 tak, by zapewnić systemową ochronę wykrywającą programy, które zostały zmodyfikowane.
- Ograniczyć i kontrolować używanie komendy OVRMSGF (Override with Message File).
- Ogranicz uprawnienia do danych produkcyjnych.

6. Modele autoryzacji

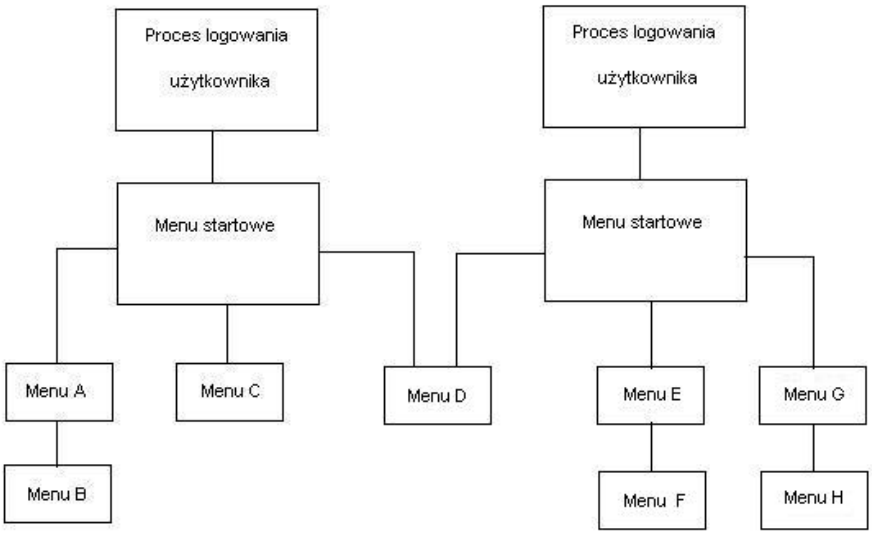
W celu ułatwienia kontroli dostępu do danych, proponuje się przyjąć jeden z kilku modeli autoryzacji. Te modele nie powinny być traktowane jako modele bezpieczeństwa systemu, ale raczej jako szkielety pomagające zbudować odpowiedni model bezpieczeństwa.

Model 1

Bardzo przejrzysty jest model, w którym użytkownik w swej pracy wykorzystuje tylko i wyłącznie menu. Bezpieczeństwo jest osiągane w prosty sposób: poprzez przedstawienie użytkownikowi menu z funkcjami, które potrzebuje do swej pracy. Nie ma on możliwości korzystania z linii komend oraz uruchamiania programów, które nie są uwzględnione w menu.

Stosując model przedstawiony na rys. 3, należy zapewnić każdemu użytkownikowi indywidualnie zaprojektowane menu startowe tylko z tymi opcjami, które potrzebuje dany użytkownik. Ułatwia to i przyspiesza pracę użytkownikowi, który jest zwolniony od wpisywania i pamiętania trudnych nazw programów. Podstawowym problemem w tym modelu jest szybko wzrastająca liczba menu, która sprawia problem w zarządzaniu nimi i kontrolowaniu ich praw dostępu.

Ten model można stosować do grupy użytkowników, których praca nie wymaga zbyt wielu opcji oraz opcje te nie ulegają zmianie.



Rys. 3. Model oparty na menu

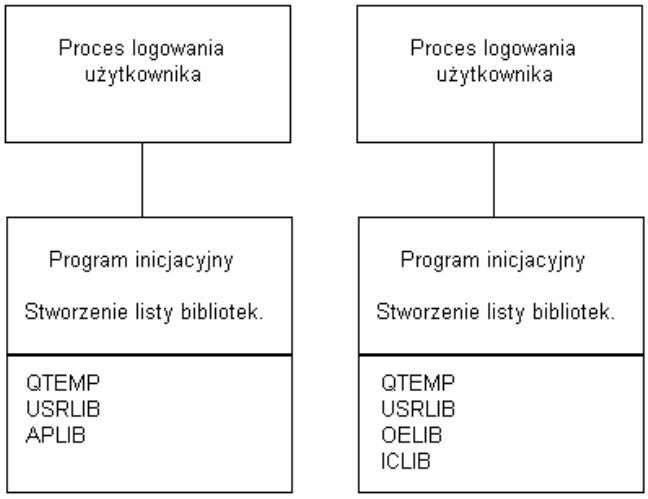
Model 2

Model ten wykorzystuje autoryzacje do bibliotek i katalogów indywidualnie dla każdego profilu użytkownika. Użytkownik ma dostęp tylko do bibliotek (minimalne środowisko pracy), które są mu niezbędne do pracy, natomiast nie ma dostępu do innych bibliotek.

Kluczem do zaimplementowania tego modelu jest to, by każdy użytkownik miał prawidłowo nadane uprawnienia do bibliotek, które są mu niezbędne. Należy się także upewnić, czy program inicjacyjny użytkownika oraz opis wykonywania zadań użytkownika (job description) dostarcza użytkownikowi prawidłową listę bibliotek.

Proponuje się podzielić biblioteki według funkcji spełnianych przez znajdujące się w nich aplikacje.

Wadą tego modelu jest to, iż daje on dostęp do całej biblioteki, nawet jeśli użytkownik potrzebuje dostępu tylko do jednego lub dwóch obiektów w danej bibliotece. Pomocne w takim przypadku może okazać się zastosowanie menu lub odpowiednio napisanych aplikacji bezpieczeństwa.



Rys. 4. Model zapewniający minimalne środowisko pracy

Model 3

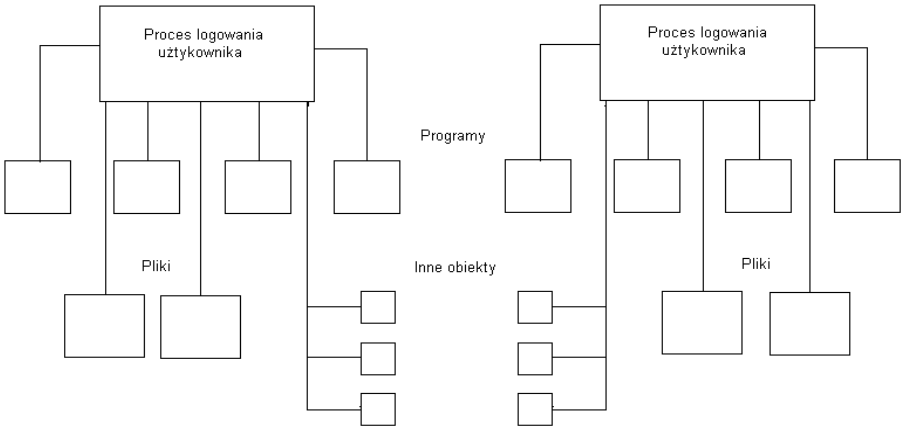
W modelu tym zaproponowano, by uprawnienia były nadawane indywidualnie do każdego obiektu w systemie. Na rys. 5 przedstawiono schemat tego modelu.

Ten model wymaga, by administrator w pełni rozumiał jak funkcjonuje system i mechanizm autoryzacji. W modelu pierwszym i drugim, domyślna autoryzacja (public authority) powinna być ustawiona na co najmniej *CHANGE, natomiast w tym modelu jest to niepotrzebne, co sprawia, że implementacja tego modelu czyni nasz system bardziej bezpieczny niż zastosowanie modelu 1 lub 2.

SPRAWDZENIE:

Określ model użytkowany w Twojej firmie:

- MODEL 1 TAK ☐ NIE ☐
- MODEL 2 TAK ☐ NIE ☐
- MODEL 3 TAK ☐ NIE ☐



Rys. 5. Model, w którym uprawnienia są nadawane indywidualnie do każdego obiektu

7. Podsumowanie

Elementy bezpieczeństwa AS/400, które przedstawiono w artykule, są tylko „kroplą w morzu" – zamiarem autora było krótkie przybliżenie tych zagadnień. Żaden system teleinformatyczny nie jest całkowicie bezpieczny. Są systemy bardziej lub mniej bezpieczne. AS/400 zalicza się do klasy systemów wiodących pod względem bezpieczeństwa. Paradoksalnie jednym z elementów, który to gwarantuje, jest bardzo nikła znajomość tego systemu w środowisku informatycznym, co jest spowodowane kosztami komputera jak i polityką producenta, który nie publikuje dokumentacji systemu.

Literatura:

[1] IBM MANUAL AS/400 : *Security – Reference* (SC41-5302-03).
[2] IBM MANUAL AS/400 : *Tips and Tools for Securing Your AS/400* (SC41-5300-03).
[3] IBM MANUAL AS/400 : *Ochrona – Podstawy* (SA12-7279-00).
[4] IBM MANUAL AS/400 : *Podstawowe czynności związane z uruchamianiem i działaniem systemu* (SA12-7268-03).

Recenzent: dr inż. Krzysztof Liderman

Praca wpłynęła do redakcji 20.10.2002