System i

Bezpieczeństwo
Podpisywanie obiektów i weryfikowanie podpisów

Wersja 6 wydanie 1

System i

Bezpieczeństwo
Podpisywanie obiektów i weryfikowanie podpisów

Wersja 6 wydanie 1

Uwaga

Przed skorzystaniem z tych informacji oraz z produktu, którego dotyczą, należy przeczytać informacje zawarte w sekcji “Uwagi”, na stronie 49.

To wydanie dotyczy systemu operacyjnego IBM i5/OS (numer produktu 5761-SS1) wersja 6, wydanie 1, modyfikacja 0, a także wszystkich kolejnych wydań i modyfikacji, chyba że w nowych wydaniach zostanie określone inaczej. Wersja ta nie działa na wszystkich modelach komputerów z procesorem RISC ani na modelach z procesorem CISC.

Spis treści

Podpisywanie obiektów i weryfikowanie podpisów

Informacje dotyczące ochrony opartej na podpisywaniu obiektów i weryfikowaniu podpisów w systemie i5/OS, którą można wykorzystać w celu zapewnienia integralności obiektów. Opis korzystania z jednej z kilku metod oferowanych przez system i5/OS w celu tworzenia podpisów cyfrowych obiektów, aby zidentyfikować źródło obiektu i umożliwić wykrycie zmian obiektu. Ponadto opis metody poprawienia ochrony systemu w wyniku weryfikowania podpisów cyfrowych obiektów, w tym obiektów systemu operacyjnego, aby wykryć, czy zawartość obiektu uległa zmianie od czasu jego podpisania.

Podpisywanie obiektów i weryfikowanie podpisów stanowią elementy ochrony systemu i służą sprawdzeniu integralności różnorodnych obiektów. Klucz prywatny certyfikatu cyfrowego służy do podpisania obiektu, zaś certyfikat (zawierający odpowiadający mu klucz publiczny) do weryfikowania podpisu cyfrowego. Podpis cyfrowy zapewnia integralność czasu i zawartości podpisywanego obiektu. Podpis zapewnia dowód zarówno autentyczności, jak i autoryzacji. Można go użyć do sprawdzenia pochodzenia i do wykrycia fałszerstwa. Podpisując obiekt identyfikuje się jego źródło i zapewnia możliwość wykrycia zmian w tym obiekcie. Podczas weryfikowania podpisu obiektu można określić, czy od czasu podpisania zawartość obiektu uległa zmianie. Można także zweryfikować źródło podpisu, aby upewnić się co do pochodzenia obiektu.

Aby wdrożyć podpisywanie obiektów i weryfikowanie podpisów, można skorzystać z następujących możliwości:

- funkcje API do programowego podpisywania obiektów i weryfikowania podpisów,
- program Digital Certificate Manager do podpisywania obiektów i do przeglądania i weryfikowania podpisów na obiektach,
- Centrum Zarządzania programu iSeries Navigator do podpisywania obiektów wchodzących w skład pakietów rozpowszechnianych do innych systemów,
- komendy CL, takie jak Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG), do weryfikowania podpisów.

Więcej o metodach podpisywania obiektów i o ich wpływie na udoskonalenie strategii ochrony można przeczytać w artykułach:

Uwaga: Korzystając z przykładów kodu, użytkownik wyraża zgodę na warunki zapisane w sekcji “Licencja na kod oraz Informacje dotyczące kodu” na stronie 47.

Co nowego w wersji V6R1

Informacje na temat zmian w kolekcji tematów Podpisywanie obiektów i weryfikowanie podpisów.

Weryfikowanie integralności funkcji kontrolera kodu

Począwszy od wersji V6R1 istnieje możliwość weryfikowania Licencjonowanego Kodu Wewnętrznego (LIC) za pomocą funkcji API Check System (QydoCheckSystem) lub komendy Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG).

Znajdowanie nowych lub zmienionych informacji

Aby ułatwić określenie obszarów, w których zostały wprowadzone zmiany techniczne, w Centrum informacyjnym zastosowano:

- symbol » służący do zaznaczania początku nowego lub zmienionego fragmentu;
- symbol « służący do zaznaczania końca nowego lub zmienionego fragmentu.

Nowe i zmienione informacje w plikach PDF mogą być oznaczone symbolem | na lewym marginesie.

Więcej informacji na temat zmian i nowości w bieżącej wersji zawiera Wiadomość dla użytkowników.

Plik PDF z informacjami na temat podpisywania obiektów i weryfikowania podpisów

Poniższe informacje umożliwiają wydrukowanie pliku PDF zawierającego cały temat dotyczący podpisywania obiektów i weryfikowania podpisów w systemie operacyjnym i5/OS.

Aby otworzyć lub pobrać wersję dokumentu w formacie PDF, kliknij odsyłacz Podpisywanie obiektów i weryfikowanie podpisów (wielkość pliku: 605 KB).

Zapisywanie plików PDF:

Aby zapisać plik PDF na stacji roboczej w celu jego wyświetlenia lub wydrukowania, wykonaj następujące czynności:

1. Kliknij prawym przyciskiem myszy odsyłacz do pliku PDF w przeglądarce.
2. Kliknij opcję zapisania pliku PDF lokalnie.
3. Przejdź do katalogu, w którym ma zostać zapisany plik PDF.
4. Kliknij opcję **Zapisz**.

Pobieranie programu Adobe Acrobat Reader

Aby wyświetlać lub drukować pliki PDF, potrzebny jest program Adobe Acrobat Reader. Kopię programu można pobrać z serwisu WWW firmy Adobe (www.adobe.com/products/acrobat/readstep.html)  .

Pojęcia związane z podpisywaniem obiektów

Ten temat zawiera pojęcia oraz informacje uzupełniające dotyczące podpisów cyfrowych w systemie i5/OS, a także procesów podpisywania obiektów i weryfikowania podpisów w tym systemie.

Przed skorzystaniem z możliwości podpisywania obiektów i weryfikowania podpisów warto zapoznać się z niektórymi z tych informacji.

Podpisy cyfrowe

Ten temat zawiera informacje o podpisach cyfrowych w systemie operacyjnym i5/OS i zapewnianej przez nie ochronie.

System operacyjny i5/OS obsługuje certyfikaty cyfrowe umożliwiające cyfrowe ″podpisywanie″ obiektów. Podpis cyfrowy na obiekcie jest tworzony metodą kryptograficzną i działa jak osobisty podpis na dokumencie. Podpis cyfrowy stanowi świadectwo pochodzenia obiektu i daje możliwość sprawdzenia jego integralności. Właściciel certyfikatu cyfrowego ″podpisuje″ obiekt za pomocą klucza prywatnego certyfikatu. Odbiorca obiektu korzysta z klucza publicznego tego samego certyfikatu w celu deszyfrowania podpisu, co weryfikuje integralność podpisanego obiektu oraz nadawcę.

Możliwość podpisywania obiektów wspomaga tradycyjne narzędzia systemu, które pozwalają kontrolować uprawnienia użytkowników do zmiany obiektów. Tradycyjne narzędzia nie mogą jednak ochronić obiektów przed nieuprawnioną modyfikacją podczas ich przesyłania poprzez Internet lub inne sieci niechronione. Ponieważ można wykryć, czy zawartość obiektu uległa zmianie od czasu jego podpisania, można również łatwo określić, czy w takich przypadkach można zaufać danemu obiektowi.

Podpis cyfrowy to zaszyfrowana suma kontrolna danych w obiekcie. Obiekt i jego zawartość nie są zaszyfrowane przez podpis cyfrowy; zaszyfrowana jest tylko suma kontrolna, aby uniemożliwić dokonanie bez uprawnień zmian obiektu. Chcąc się upewnić, że obiekt nie został zmieniony podczas przesyłania i że pochodzi z akceptowanego, legalnego źródła, należy użyć klucza publicznego certyfikatu wykorzystanego do podpisu, aby sprawdzić autentyczność podpisu

cyfrowego. Jeśli podpis nie będzie zgodny, może to oznaczać, że dane zostały zmienione. W takim przypadku odbiorca może, zamiast użyć obiektu, skontaktować się z nadawcą i poprosić o przesłanie kopii podpisanego obiektu.

Podpis na obiekcie reprezentuje system, który podpisał ten obiekt, a nie konkretnego użytkownika tego systemu (chociaż użytkownik musi mieć odpowiednie uprawnienia, aby użyć certyfikatu do podpisania obiektu).

Jeśli użycie certyfikatów cyfrowych mieści się w ramach zidentyfikowanych potrzeb i przyjętych strategii bezpieczeństwa, należy jeszcze rozstrzygnąć, czy powinno się używać certyfikatów publicznych, czy wystawiać certyfikaty prywatne. W przypadku dystrybucji obiektów do użytkowników publicznych należy rozważyć podpisywanie obiektów za pomocą certyfikatów z ogólnie znanego publicznego ośrodka certyfikacji CA. Certyfikaty publiczne pozwalają innym łatwo i tanio zweryfikować podpisy złożone na wysyłanych im obiektach. Jeśli jednak zamierza się rozpowszechniać obiekty wyłącznie w ramach własnej organizacji, wygodniejsze może być użycie programu Digital Certificate Manager (DCM) w celu poprowadzenia własnego ośrodka certyfikacji i wystawiania prywatnych certyfikatów do podpisywania obiektów. Korzystanie z prywatnych certyfikatów lokalnego ośrodka CA jest tańsze niż certyfikaty pochodzące od powszechnie znanego publicznego ośrodka CA.

Rodzaje podpisów cyfrowych

Począwszy od wersji V5R2 można podpisywać obiekty typu komenda (*CMD) i wybierać pomiędzy dwoma typami podpisów tych obiektów: rdzenia obiektu i całego obiektu.

- **Podpisy całych obiektów** Ten typ podpisu zawiera wszystkie ważne bajty obiektu oprócz kilku mniej istotnych.
- **Podstawowe podpisy obiektów** Ten typ podpisu obejmuje najważniejsze bajty obiektu *CMD. Nie obejmuje natomiast tych bajtów, które są przedmiotem najczęstszych zmian. Umożliwia wprowadzenie pewnych zmian do komendy, bez naruszenia podpisu. To, które bajty podpisywanego rdzenia obiektu są pomijane, zależy od określonego obiektu *CMD, na przykład podpisy rdzenia nie obejmują domyślnych parametrów obiektów *CMD. Przykłady zmian, które nie wpłyną na podpis rdzenia to:
  - zmiana wartości domyślnych komendy,
  - dodanie do komendy programu sprawdzania poprawności,
  - zmiana parametru Dozwolone środowisko wykonania,
  - zmiana parametru Zezwolenie na ograniczenie użytkowników.

  Pojęcia pokrewne

  “Obiekty do podpisywania”
  Ten temat zawiera informacje o obiektach, które użytkownik może podpisać oraz opcjach podpisywania obiektów komend (*CMD) systemu i5/OS.

  Informacje pokrewne

  Digital Certificate Manager (DCM)

Obiekty do podpisywania

Ten temat zawiera informacje o obiektach, które użytkownik może podpisać oraz opcjach podpisywania obiektów komend (*CMD) systemu i5/OS.

Niezależnie od użytych do podpisywania metod, można podpisywać cyfrowo różne typy obiektów systemu i5/OS. Podpisywać można dowolny obiekt (*STMF) przechowywany w systemowym zintegrowanym systemie plików, poza obiektami przechowywanymi w bibliotece. Jeśli do obiektu dołączony jest program w języku Java, zostanie on również podpisany. W systemie plików QSYS.LIB można podpisywać tylko następujące obiekty: programy (*PGM), programy serwisowe (*SRVPGM), moduły (*MODULE), pakiety SQL (*SQLPKG), *FILE (tylko zbiory składowania) i komendy (*CMD).

Aby obiekt mógł być podpisany, musi znajdować się w systemie lokalnym. Na przykład podczas pracy z systemem Windows 2000 na serwerze Integrated xSeries Server for System i zintegrowany system plików udostępnia system plików QNTC. Katalogi w tym systemie plików nie są uważane za lokalne, gdyż zawierają pliki należące do systemu operacyjnego Windows 2000. Nie można też podpisywać pustych obiektów lub obiektów skompilowanych dla wersji systemu wcześniejszych niż V5R1.

Podpisywanie obiektów typu komenda (*CMD)

Do podpisywania obiektów *CMD można wybrać jeden z dwóch typów podpisów. Można wybrać pomiędzy podpisaniem całego obiektu lub tylko jego rdzennej części. Jeśli podpisywany jest cały obiekt, podpis obejmuje wszystkie, w tym część mniej ważnych, bajty obiektu. Podpis całego obiektu obejmuje elementy zawarte w podpisie rdzenia obiektu.

Jeśli wybrane zostanie podpisywanie tylko rdzenia obiektu, chronione przez podpis będą tylko najważniejsze bajty, zaś bajty, które ulegają częstszym zmianom nie zostaną podpisane. Które bajty zostaną niepodpisane zależy od obiektu *CMD, ale można w podpisie zawrzeć między innymi określenie trybu, w którym obiekt jest poprawny lub określenie, czy obiekt ma prawo być uruchamiany. Na przykład podpisy rdzenia nie zawierają wartości domyślnych parametrów obiektów *CMD. Umożliwia to wprowadzenie pewnych zmian do komendy, bez naruszenia jej podpisu. Przykłady zmian, które nie wpłyną na ten typ podpisu to:

- zmiana wartości domyślnych komendy,
- dodanie do komendy programu sprawdzania poprawności,
- zmiana parametru Dozwolone środowisko wykonania,
- zmiana parametru Zezwolenie na ograniczenie użytkowników.

Tabela przedstawia, które dokładnie bajty obiektu *CMD zostaną zawarte w podpisywanym rdzeniu obiektu.

Skład podpisywanego rdzenia obiektu dla obiektów *CMD

Część obiektu	**Czy należy do podpisywanego rdzenia obiektu**
Wartości domyślne komendy zmienione przez CHGCMDDFT	Nie jest częścią podpisywanego rdzenia obiektu
Program do przetwarzania komend z biblioteką	Zawsze jest częścią podpisywanego rdzenia obiektu
Zbiór źródłowy REXX z biblioteką	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Podzbiór źródłowy REXX	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Środowisko komend REXX z biblioteką	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Nazwa programu obsługi wyjścia REXX, biblioteki i kod wyjścia	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Program sprawdzania poprawności z biblioteką	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Poprawny tryb dla komendy	Nie jest częścią podpisywanego rdzenia obiektu
Dozwolone środowisko wykonania	Nie jest częścią podpisywanego rdzenia obiektu
Zezwolenie na ograniczenie użytkowników	Nie jest częścią podpisywanego rdzenia obiektu
Półka pomocy	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Panel grupowy pomocy z biblioteką	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu

Część obiektu	Czy należy do podpisywanego rdzenia obiektu
Identyfikator pomocy	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Indeks wyszukiwania pomocy z biblioteką	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Biblioteka bieżąca	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Biblioteka produktu	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Program przesłaniający podpowiedzi i biblioteka	Włączany do podpisu, jeśli został określony dla komendy w momencie podpisywania, w przeciwnym razie nie jest częścią podpisywanego rdzenia obiektu
Tekst (opis)	Nie jest częścią podpisywanego rdzenia obiektu ani podpisu całego obiektu, gdyż nie jest przechowywany w obiekcie
Włączenie interfejsu GUI	Nie jest częścią podpisywanego rdzenia obiektu

Pojęcia pokrewne

“Podpisy cyfrowe” na stronie 2
Ten temat zawiera informacje o podpisach cyfrowych w systemie operacyjnym i5/OS i zapewnianej przez nie ochronie.

Przetwarzanie podpisywania obiektów

Ten temat zawiera informacje o przebiegu procesu podpisywania obiektów w systemie, w którym działa system operacyjny i5/OS. Ponadto opisano w nim parametry, które mogą zostać ustawione dla tego procesu.

Podczas podpisywania obiektów można dla tego procesu określić następujące opcje.

Przetwarzanie po wystąpieniu błędu
Pozwala określić, jakiego typu przetwarzania po wystąpieniu błędu powinna używa aplikacja podczas tworzenia podpisów na więcej niż jednym obiekcie. Do wyboru jest zatrzymanie podpisywania obiektów po wystąpieniu błędu lub kontynuacja podpisywania na pozostałych obiektach.

Podwójny podpis obiektu
Pozwala określić, w jaki sposób aplikacja obsługuje proces podpisywania, jeśli obiekt jest podpisywany ponownie. Do wyboru jest zostawienie pierwotnego podpisu lub zastąpienie go nowym.

Obiekty w podkatalogach
Pozwala określić, w jaki sposób aplikacja obsługuje podpisywane obiekty znajdujące się w podkatalogach. Do wyboru jest indywidualne podpisywanie przez aplikację obiektów we wszystkich podkatalogach lub podpisywanie tylko w katalogu głównym z wyłączeniem podkatalogów.

Zasięg podpisu obiektu
Podczas podpisywania obiektów *CMD można określić, czy podpisywany ma być cały obiekt, czy tylko jego rdzenna część.

Weryfikowanie podpisów

Ten temat zawiera informacje o procesie weryfikowania podpisów obiektów w systemie operacyjnym i5/OS. Ponadto opisano w nim parametry, które mogą zostać ustawione dla tego procesu.

Przy weryfikowaniu podpisów można określić następujące opcje.

Przetwarzanie po wystąpieniu błędu
Pozwala określić, jakiego typu przetwarzania po wystąpieniu błędu powinna używa aplikacja podczas weryfikowania podpisów na więcej niż jednym obiekcie. Do wyboru jest zatrzymanie weryfikowania podpisów po wystąpieniu błędu lub kontynuacja weryfikacji na pozostałych obiektach.

Obiekty w podkatalogach
Pozwala określić, w jaki sposób aplikacja obsługuje podpisy weryfikowane na obiektach znajdujących się w podkatalogach. Do wyboru jest indywidualne weryfikowanie przez aplikację podpisów na obiektach we wszystkich podkatalogach lub weryfikowanie podpisów tylko dla obiektów w katalogu głównym z wyłączeniem podkatalogów.

Weryfikowanie podpisu rdzenia a weryfikowanie podpisu całości
Istnieją pewne reguły systemowe określające, w jaki sposób system obsługuje podpisy rdzenia lub całego obiektu podczas procesu weryfikacji. Reguły są następujące:

- Jeśli na obiekcie nie ma żadnego podpisu, proces weryfikujący zgłasza, że obiekt nie jest podpisany, a następnie weryfikuje kolejne przetwarzane obiekty.
- Jeśli obiekt został podpisany przez zaufane źródło (IBM), podpis musi być zgodny, w innym wypadku proces weryfikacji nie powiedzie się. Jeśli podpis jest zgodny, to proces weryfikacji trwa nadal. Podpis jest zaszyfrowaną sumą danych w obiekcie, dlatego zakłada się, że podpis będzie zgodny, jeśli dane w obiekcie podczas weryfikowania są zgodne z danymi w obiekcie w czasie jego podpisywania.
- Jeśli obiekt ma jakikolwiek zaufany podpis całego obiektu (zaufanie określane w oparciu o certyfikaty znajdujące się w bazie certyfikatów *SIGNATUREVERIFICATION), to przynajmniej jeden z tych podpisów musi być zgodny, inaczej proces weryfikacji nie powiedzie się. Jeśli przynajmniej jeden z podpisów całego obiektu jest zgodny, to proces weryfikacji trwa nadal.
- Jeśli obiekt ma jakikolwiek zaufany podpis rdzenia obiektu, to przynajmniej jeden z podpisów musi być zgodny z certyfikatem z bazy certyfikatów *SIGNATUREVERIFICATION, inaczej proces weryfikacji nie powiedzie się. Jeśli przynajmniej jeden z podpisów rdzenia obiektu jest zgodny, to proces weryfikacji trwa nadal.

Funkcja weryfikowania integralności kontrolera kodu

Ten temat zawiera informacje o weryfikowaniu integralności kontrolera kodu, który służy do weryfikowania integralności systemów działających pod kontrolą systemu operacyjnego i5/OS.

| Wersja V5R2 systemu i5/OS dostarczana jest z funkcją kontrolera kodu, której można użyć do weryfikowania
| integralności podpisanych obiektów w systemie, w tym całego kodu systemu operacyjnego dostarczonego i
| podpisanego przez firmę IBM dla danego systemu. Począwszy od wersji V5R3 dostępna jest nowa funkcja API Check
| System, za pomocą której można weryfikować integralność samej funkcji sprawdzania kodu oraz kluczowych
| obiektów systemu operacyjnego. Licencjonowany Kod Wewnętrzny (kod LIC) jest obecnie podpisywany przez firmę
| IBM i użytkownik może go zweryfikować za pomocą funkcji API Check System (QydoCheckSystem) lub komendy
| Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG).

Funkcja API Check System (QydoCheckSystem) umożliwia zweryfikowanie integralności systemu i5/OS. Używa się jej do weryfikowania programów (*PGM) i programów usługowych (*SRVPGM) oraz wybranych komend (*CMD) obiektów w bibliotece QSYS. Ponadto funkcja API Check System testuje komendy Odtworzenie obiektu (Restore Object - RSTOBJ), Odtworzenie biblioteki (Restore Library - RSTLIB), Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG) oraz funkcję API Verify Object. Ten test zapewnia, że powyższe komendy oraz funkcja API Verify Object zgłoszą właściwe błędy sprawdzania podpisu, na przykład wtedy, gdy system dostarcza obiekt nie podpisany lub posiadający nieważny podpis.

Funkcja API Check System zapisuje komunikaty o błędach w przypadku niepowodzenia weryfikacji oraz w razie wystąpienia innych błędów i niepowodzeń weryfikacji w protokole zadania. Można również określić dwie dodatkowe metody raportowania błędów, w zależności od tego jak zostaną ustawione następujące opcje:

- Jeśli wartość systemowa QAUDLVL jest ustawiona na *AUDFAIL, wtedy funkcja API Check System generuje rekord kontroli, aby zgłosić każde niepowodzenie i błąd znaleziony przez komendy Odtworzenie obiektu (Restore Object - RSTOBJ), Odtworzenie biblioteki (Restore Library - RSTLIB) i Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG).
- Jeśli użytkownik określi, że funkcja API Check System używa pliku wynikowego w zintegrowanym systemie plików, wtedy funkcja API albo tworzy ten plik, jeśli nie istnieje, albo dołącza do pliku zgłoszone wszystkie niepowodzenia i błędy znalezione przez tę funkcję API.

Zadania pokrewne

"Weryfikowanie integralności funkcji kontrolera kodu" na stronie 43
Weryfikowanie integralności funkcji kontrolera kodu używanej do weryfikowania integralności systemu i5/OS.

Scenariusze podpisywania obiektów

Ten temat zawiera scenariusze przedstawiające typowe sytuacje związane z używaniem funkcji podpisywania obiektów i weryfikowania podpisów w systemie i5/OS. W każdym scenariuszu przedstawiono także zadania konfiguracyjne, które należy wykonać, aby go zaimplementować zgodnie z opisem.

System udostępnia kilka różnych metod podpisywania obiektów i weryfikowania podpisów. Sposób podpisywania obiektów i przetwarzanie podpisanych obiektów zależy od wymagań i celów związanych z firmą i realizowaną strategią ochrony. Czasami potrzeba tylko zweryfikować podpis na obiekcie, aby upewnić się, że nie została naruszona integralność obiektu. Kiedy indziej zaś podpisać obiekty wysyłane do innych użytkowników lub systemów. Podpisanie obiektów umożliwia innym określenie pochodzenia obiektów i sprawdzenie ich integralności.

Wybór metody zależy od wielu czynników. Przedstawione w tym artykule scenariusze opisują niektóre z podstawowych celów podpisywania obiektów i weryfikowania podpisów realizowanych w typowej firmie. Każdy scenariusz opisuje także wymagania wstępne i zadania, które trzeba wykonać, aby go zaimplementować zgodnie z opisem. Aby określić, jak korzystać z możliwości podpisywania obiektów, dopasowując je do wymagań firmy i realizowanej strategii ochrony, należy przejrzeć poniższe scenariusze:

Scenariusz: podpisywanie obiektów i weryfikowanie podpisów za pomocą programu DCM

W scenariuszu przedstawiono przedsiębiorstwo, które chce podpisywać aplikacje dostępne na jego publicznym serwerze WWW, aby łatwo określić, czy zostały w nich wprowadzone nieuprawnione zmiany. Wykorzystanie programu Digital Certificate Manager (DCM) jako podstawowej metody podpisywania obiektów i weryfikowania podpisów obiektów w systemie operacyjnym i5/OS zostało opisane w oparciu o potrzeby przedsiębiorstwa i jego cele związane z bezpieczeństwem.

Sytuacja

Jako administrator przedsiębiorstwa MojaFirma S.A. użytkownik jest odpowiedzialny za administrowanie dwoma systemami przedsiębiorstwa. Jednym z systemów jest publiczny serwis WWW przedsiębiorstwa. Zawartość tego serwisu WWW tworzona jest w wewnętrznym, produkcyjnym systemie przedsiębiorstwa, a pliki i obiekty programów są przenoszone po ich zweryfikowaniu na publiczny serwer WWW.

Publiczny serwer WWW przedsiębiorstwa zawiera serwis WWW z ogólnymi informacjami o przedsiębiorstwie. Znajdują się tam też formularze, które klienci wypełniają podczas rejestrowania produktów, żądania informacji o produkcie, zawiadomienia o aktualizacji produktu, informacje o miejscach dystrybucji produktu itp. Zachodzi obawa, że programy cgi-bin udostępniające te formularze będą słabym punktem zabezpieczenia, ponieważ można je modyfikować. Dlatego należy stworzyć możliwość sprawdzania integralności tych programów i wykrywania zmian wprowadzonych przez nieuprawnione osoby. Aby osiągnąć ten cel ochrony, wystarczy podpisać cyfrowo te obiekty.

Przegląd możliwości podpisywania obiektów w systemie i5/OS pokazuje, że jest kilka metod, z których można skorzystać w celu podpisywania obiektów i weryfikowania podpisów obiektów. Ponieważ użytkownik odpowiada za administrowanie małą liczbą systemów i nie przewiduje konieczności częstego podpisywania obiektów, decyduje się na

wykorzystanie programu Digital Certificate Manager (DCM) w celu wykonywania tych zadań. Ponadto chcesz utworzyć lokalny ośrodek CA i do podpisywania obiektów używać certyfikatów prywatnych. Wykorzystanie certyfikatów prywatnych, wydawanych przez lokalny ośrodek CA, ogranicza wydatki związane z używaniem tej technologii ochrony, gdyż nie trzeba kupować certyfikatu od ogólnie znanego ośrodka CA.

Przykład ten jest praktycznym wprowadzeniem do konfigurowania i korzystania z podpisywania obiektów dla małej liczby systemów.

Zalety scenariusza

Scenariusz ten ma następujące zalety:

- Podpisywanie obiektów umożliwia sprawdzanie integralności obiektów narażonych na atak i łatwe określenie, czy zostały one zmienione od czasu podpisania. Pozwala zmniejszyć nakłady na przyszłe rozwiązywanie i śledzenie problemów w aplikacji lub w systemie.
- Wykorzystanie do podpisywania obiektów i weryfikowania podpisów obiektów graficznego interfejsu użytkownika programu DCM umożliwia szybkie i łatwe wykonywanie tych zadań.
- Program DCM zastosowany do podpisywania obiektów i weryfikowania podpisów obiektów pozwala zredukować czas przeznaczony na zrozumienie i wykorzystanie podpisywania obiektów jako części strategii ochrony.
- Wykorzystanie do podpisywania obiektów certyfikatu wystawionego przez prywatny ośrodek certyfikacji obniża koszty wprowadzenia podpisywania obiektów.

Cele

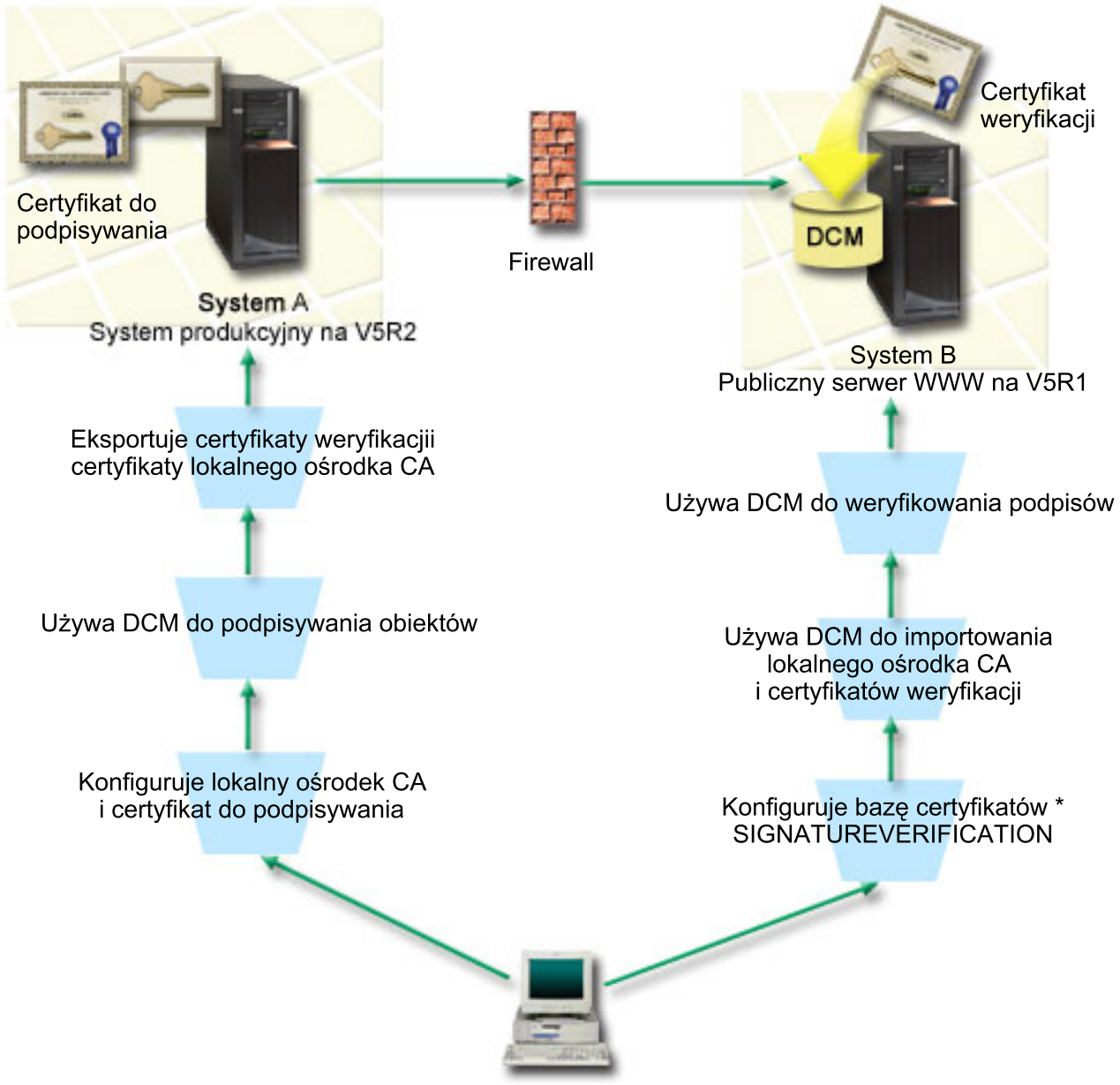
W ramach scenariusza założono, że użytkownik zamierza podpisywać cyfrowo wrażliwe obiekty, takie jak generujące formularze programy cgi-bin znajdujące się na publicznym serwerze przedsiębiorstwa. Jako administrator systemu przedsiębiorstwa MojaFirma chcesz do podpisywania tych obiektów i do weryfikowania ich podpisów użyć programu Digital Certificate Manager(DCM).

Główne założenia scenariusza są następujące:

- Aplikacje przedsiębiorstwa i inne narażone na atak obiekty, które znajdują się na dostępnym publicznie serwerze WWW (system B), muszą być podpisane certyfikatem pochodzącym z lokalnego ośrodka CA, aby ograniczyć koszty podpisywania aplikacji.
- Administratorzy systemu i inni wyznaczeni użytkownicy muszą mieć możliwość prostego weryfikowania podpisów cyfrowych w systemach, aby sprawdzić źródło i autentyczność obiektów podpisanych przez przedsiębiorstwo. Aby osiągnąć postawiony cel, każdy system musi mieć w bazie certyfikatów *SIGNATUREVERIFICATION zarówno kopię certyfikatu przedsiębiorstwa do weryfikowania podpisów, jak i certyfikat lokalnego ośrodka CA.
- Dzięki weryfikowaniu podpisów aplikacji przedsiębiorstwa i innych obiektów, administratorzy i inni użytkownicy mogą wykryć, czy zawartość obiektu została zmieniona od czasu jego podpisania.
- Administrator systemu musi korzystać z programu DCM do podpisywania obiektów, ponadto zarówno administrator systemu, jak i inni użytkownicy, powinni korzystać z programu DCM do weryfikowania podpisów obiektów.

Informacje szczegółowe

Poniższy rysunek przedstawia proces podpisywania obiektu i weryfikowania podpisu według scenariusza:



Rysunek ilustruje następujące punkty scenariusza:

System A

- System A to serwer System i, na którym działa system operacyjny OS/400 Wersja 5, Wydanie 2 (V5R2).
- System A to wewnętrzny system produkcyjny przedsiębiorstwa i platforma programistyczna dla publicznego serwera WWW System i (System B).
- W systemie A zainstalowany jest produkt Cryptographic Access Provider 128-bit for System i (5722–AC3).
- W systemie A zainstalowano i skonfigurowano produkty Digital Certificate Manager (opcja 34) oraz IBM HTTP Server (5722–DG1).
- System A funkcjonuje jako lokalny ośrodek CA i w tym systemie znajdują się certyfikaty podpisujące obiekty.

- System A podpisuje obiekty za pomocą programu DCM i jest podstawowym systemem podpisującym obiekty dla publicznych aplikacji i innych obiektów przedsiębiorstwa.
- System A został skonfigurowany w taki sposób, aby umożliwiał weryfikowanie podpisów.

System B

- System B to serwer System i, na którym działa system operacyjny OS/400 Wersja 5, Wydanie 1 (V5R1).
- System B jest zewnętrznym, publicznym serwerem WWW, znajdującym się poza siecią przedsiębiorstwa chronioną przez firewall.
- W systemie B zainstalowany jest produkt Cryptographic Access Provider 128-bit (5722–AC3).
- W systemie B zainstalowano i skonfigurowano produkty Digital Certificate Manager (opcja 34) oraz IBM HTTP Server (5722–DG1).
- System B nie funkcjonuje jako lokalny ośrodek CA, ponadto nie podpisuje obiektów.
- System B ma włączone weryfikowanie podpisów przy użyciu programu DCM tworzącego bazę certyfikatów *SIGNATUREVERIFICATION i importującego niezbędne certyfikaty weryfikacji i lokalnego ośrodka CA.
- Program DCM używany jest do weryfikowania podpisów na obiektach.

Wymagania wstępne i założenia

Scenariusz zależy od spełnienia następujących założeń i wymagań wstępnych:

1. Wszystkie systemy spełniają wymagania w zakresie instalowania i używania programu Digital Certificate Manager (DCM).
2. W żadnym systemie nie był wcześniej konfigurowany ani używany program DCM.
3. We wszystkich systemach zainstalowano najnowszą wersję programu licencjonowanego Cryptographic Access Provider 128-bit (5722-AC3).
4. Dla wartości systemowej Weryfikowanie podpisów obiektów podczas odtwarzania (VeriFY OBJect signatures during ReSTore - QVFYOBJRST) określono domyślną wartość 3 we wszystkich systemach objętych scenariuszem i nie została ona zmieniona. Ustawienie domyślne daje gwarancję, że system będzie mógł zweryfikować podpisy po odtworzeniu podpisanych obiektów.
5. Aby administrator systemu A mógł podpisywać obiekty, musi mieć uprawnienia specjalne *ALLOBJ lub jego profil użytkownika musi być uprawniony do korzystania z aplikacji podpisującej obiekty.
6. Administrator systemu lub inna osoba, która tworzy bazę certyfikatów w programie DCM, musi mieć uprawnienia specjalne *SECADM i *ALLOBJ.
7. Administrator systemu lub inni użytkownicy pozostałych systemów muszą mieć uprawnienie specjalne *AUDIT w celu weryfikowania podpisów obiektów.

Czynności konfiguracyjne

Poniżej przedstawiono dwa zestawy zadań, które należy wykonać, aby wdrożyć ten scenariusz. Jeden zestaw zadań pozwala skonfigurować system A jako lokalny ośrodek CA i podpisywać oraz weryfikować podpisy obiektów. Drugi zestaw zadań pozwala skonfigurować system B w celu weryfikowania utworzonych przez system A podpisów obiektów.

Więcej informacji dotyczących wykonania tych czynności zamieszczono w szczegółowych opisach scenariuszy poniżej.

Zadania dotyczące systemu A

Aby zgodnie ze scenariuszem system A stał się lokalnym ośrodkiem CA i mógł podpisywać obiekty oraz weryfikować podpisy obiektów, należy wykonać następujące czynności:

1. Spełnienie wszystkich wymagań wstępnych koniecznych do zainstalowania i skonfigurowania wszystkich potrzebnych produktów serwera System i.
2. Używanie programu DCM w celu utworzenia lokalnego ośrodka CA wystawiającego certyfikat podpisujący obiekt.

3. Używanie programu DCM w celu tworzenia definicji aplikacji.
4. Używanie programu DCM w celu przypisana certyfikatu do definicji aplikacji podpisującej obiekty.
5. Używanie programu DCM w celu podpisania obiektów programów cgi-bin.
6. Używanie programu DCM w celu eksportowania certyfikatów, których inne systemy muszą użyć do zweryfikowania podpisów obiektów. Konieczne jest wyeksportowanie do pliku zarówno kopii certyfikatu lokalnego ośrodka CA, jak i kopii certyfikatu do podpisywania obiektów, jako certyfikatu do weryfikowania podpisów.
7. Przesłanie plików certyfikatów na publiczny serwer WWW (system B), dzięki czemu użytkownik i inne osoby mogą weryfikować podpisy utworzone przez system A.

Zadania dotyczące systemu B

Jeśli planowane jest przywrócenie podpisanych obiektów, które zostały przesłane na publiczny serwer WWW w ramach tego scenariusza (system B), zadania związane z konfiguracją weryfikacji podpisów w systemie B należy zrealizować przed przesłaniem podpisanych obiektów. Konfigurowanie weryfikacji podpisu należy zakończyć przed rozpoczęciem weryfikowania na publicznym serwerze WWW podpisów odtworzonych podpisanych obiektów.

W systemie B należy wykonać poniższe czynności, aby zgodnie z założeniami scenariusza móc weryfikować podpisy obiektów:

1. Użyj programu Digital Certificate Manager (DCM) w celu utworzenia bazy certyfikatów *SIGNATUREVERIFICATION
2. Użyj programu DCM w celu zaimportowania certyfikatu lokalnego ośrodka CA i certyfikatu do weryfikowania podpisu
3. Użyj programu DCM w celu weryfikowania podpisów przesłanych obiektów

Informacje pokrewne

Digital Certificate Manager (DCM)

Szczegóły scenariusza: podpisywanie obiektów i weryfikowanie podpisów za pomocą programu DCM

Ten temat zawiera informacje o konfigurowaniu programu Digital Certificate Manager i podpisywaniu za jego pomocą obiektów systemu operacyjnego i5/OS.

Czynność 1: spełnienie wszystkich wymagań wstępnych

Przed wykonaniem określonych zadań konfiguracyjnych związanych z implementacją tego scenariusza należy w całości spełnić wymagania wstępne w zakresie instalowania i konfigurowania wszystkich potrzebnych produktów serwera System i.

Czynność 2: tworzenie lokalnego ośrodka certyfikacji w celu wystawienia certyfikatu do podpisywania obiektu prywatnego

Podczas tworzenia lokalnego ośrodka CA za pomocą programu Digital Certificate Manager należy wypełnić szereg formularzy. Formularze te prowadzą przez proces tworzenia ośrodka CA i inne czynności niezbędne, aby rozpocząć korzystanie z certyfikatów cyfrowych w połączeniu z protokołem SSL, czyli podpisywanie obiektów i weryfikowanie podpisów. Wprawdzie w tym scenariuszu nie trzeba konfigurować certyfikatów w połączeniu z protokołem SSL, ale w celu skonfigurowania systemu do podpisywania obiektów należy wypełnić wszystkie formularze.

Aby użyć programu DCM w celu utworzenia i stosowania lokalnego ośrodka CA, wykonaj następujące czynności: po utworzeniu lokalnego ośrodka CA i certyfikatu podpisującego obiekt, przed podpisywaniem obiektów należy zdefiniować korzystającą z certyfikatu aplikację podpisującą obiekty.

1. Uruchom program DCM. Więcej informacji zawiera temat Uruchamianie programu DCM.
2. W ramce nawigacji programu DCM wybierz **Tworzenie ośrodka certyfikacji**, aby wyświetlić szereg formularzy.

Uwaga: W przypadku pojawienia się pytań dotyczących określonego formularza w tym zadaniu należy kliknąć przycisk znaku zapytania (**?**) u góry strony, aby uzyskać dostęp do systemu pomocy.

3. Wypełnij wszystkie formularze zadań. Po zakończeniu tego zadania, wykonaj poniższe czynności:
   a. Wprowadź informacje identyfikujące lokalny ośrodek CA.
   b. Zainstaluj certyfikat lokalnego ośrodka CA w swojej przeglądarce, aby oprogramowanie mogło go rozpoznać i sprawdzać poprawność certyfikatów wystawionych przez ten ośrodek.
   c. Zdefiniuj strategię lokalnego ośrodka CA.
   d. Użyj nowego lokalnego ośrodka CA do wystawienia certyfikatu klienta lub serwera, z którego aplikacja będzie mogła skorzystać do połączeń z użyciem protokołu SSL.

      Uwaga: Wprawdzie opisywany scenariusz nie korzysta z tego certyfikatu, jego utworzenie jest jednak niezbędne przed użyciem lokalnego ośrodka CA do wystawienia potrzebnego certyfikatu podpisującego obiekt. Jeśli zadanie zostanie anulowane bez utworzenia certyfikatu, to należy utworzyć certyfikat podpisujący obiekt i bazę certyfikatów *OBJECTSIGNING, w której będzie on przechowywany.
   e. Wybierz aplikacje, które będą mogły używać certyfikatu serwera lub klienta w połączeniach SSL.

      Uwaga: Na potrzeby tego scenariusza nie wybieraj żadnej aplikacji, tylko kliknij **Kontynuuj**, aby wyświetlić następny formularz.
   f. Użyj nowego lokalnego ośrodka CA do wystawienia certyfikatu podpisującego obiekt, z którego aplikacja będzie mogła skorzystać do cyfrowego podpisywania obiektów. W tym podzadaniu tworzona jest baza certyfikatów *OBJECTSIGNING. Jest to baza umożliwiająca zarządzanie certyfikatami do podpisywania obiektów.
   g. Wybierz aplikacje, które powinny ufać lokalnemu ośrodkowi CA.

      Uwaga: W ramach tego scenariusza nie wybieraj żadnych aplikacji i kliknij przycisk **Kontynuuj**, aby zakończyć zadanie.

Czynność 3: tworzenie definicji aplikacji podpisującej obiekty

Po utworzeniu certyfikatu podpisującego obiekt należy za pomocą programu Digital Certificate Manager utworzyć definicję aplikacji podpisującej obiekty, która będzie używana do podpisywania obiektów. Definicja aplikacji nie musi odnosić się do istniejącej aplikacji. Tworzona definicja aplikacji powinna opisywać typ lub grupę obiektów, które zamierzasz podpisywać. Definicja jest niezbędna, żeby można było powiązać identyfikator aplikacji z certyfikatem i umożliwić proces podpisywania.

Aby utworzyć definicję aplikacji podpisującej obiekty za pomocą programu DCM, wykonaj następujące czynności:

1. W ramce nawigacyjnej kliknij opcję **Wybór bazy certyfikatów** i wybierz ***OBJECTSIGNING** jako bazę certyfikatów, która ma zostać otworzona.
2. Po wyświetleniu strony Baza certyfikatów i hasło wpisz hasło określone dla bazy certyfikatów przy jej tworzeniu i kliknij przycisk **Kontynuuj**.
3. W ramce nawigacji wybierz **Zarządzanie aplikacjami**, aby wyświetlić listę zadań.
4. Z listy zadań wybierz **Dodaj aplikację**, aby wyświetlić formularz definiowania aplikacji.
5. Wypełnij formularz i kliknij **Dodaj**.

Należy teraz przypisać certyfikat podpisujący obiekt do utworzonej aplikacji.

Czynność 4: przypisanie certyfikatu do definicji aplikacji podpisującej obiekty

Aby przypisać certyfikat do aplikacji podpisującej obiekty, wykonaj następujące czynności:

1. W ramce nawigacji programu DCM wybierz **Zarządzanie certyfikatami**, aby wyświetlić listę zadań.
2. Z listy tej wybierz **Przypisanie certyfikatu**, aby wyświetlić listę certyfikatów bieżącej bazy certyfikatów.

3. Wybierz certyfikat z listy i kliknij **Przypisz do aplikacji**, aby wyświetlić listę definicji aplikacji bieżącej bazy certyfikatów.
4. Wybierz jedną lub więcej aplikacji z listy i kliknij **Kontynuuj**. Pojawi się strona komunikatów, przedstawiająca albo potwierdzenie przypisania certyfikatu, albo informacje o błędzie, jeśli wystąpił jakiś problem.

Po zakończeniu tych czynności można korzystać z programu DCM w celu podpisywania obiektów programów, które będą wykorzystywane na publicznym serwerze WWW firmy (system B).

Czynność 5: podpisywanie obiektów programów

Aby za pomocą programu DCM podpisywać programy używane na publicznym serwerze WWW przedsiębiorstwa (system B), wykonaj następujące czynności:

1. W ramce nawigacyjnej kliknij opcję **Wybór bazy certyfikatów** i wybierz ***OBJECTSIGNING** jako bazę certyfikatów, która ma zostać otworzona.
2. Wprowadź hasło bazy certyfikatów *OBJECTSIGNING i kliknij przycisk **Kontynuuj**.
3. Po odświeżeniu widoku ramki nawigacji wybierz **Zarządzanie obiektami do podpisywania**, aby wyświetlić listę zadań.
4. Z listy zadań wybierz **Podpisanie obiektu**, aby wyświetlić listę definicji aplikacji, których można użyć do podpisywania obiektów.
5. Wybierz aplikację zdefiniowaną w ramach poprzedniej czynności i kliknij opcję **Podpisanie obiektu**. Pojawi się formularz umożliwiający podanie położenia obiektów, które mają być podpisane.
6. W wyświetlone pola wpisz pełną ścieżkę i nazwę pliku obiektu lub katalogu obiektów, które chcesz podpisać, i kliknij **Kontynuuj**. Można również wpisać położenie katalogu i kliknąć **Przeglądaj**, aby przejrzeć zawartość katalogu i wybrać obiekty do podpisu.

   Uwaga: Nazwa obiektu musi zaczynać się od ukośnika, w przeciwnym razie mogą wystąpić błędy. Do określenia części obiektów katalogu, które mają zostać podpisane, można także użyć znaków zastępczych. Te znaki to gwiazdka (*****), która zastępuje *dowolny ciąg znaków*, i znak zapytania (**?**), który zastępuje *dowolny pojedynczy znak*. Na przykład, aby podpisać wszystkie obiekty w określonym katalogu, można wpisać /mydirectory/* ; aby podpisać wszystkie programy w określonej bibliotece, można wpisać /QSYS.LIB/QGPL.LIB/*.PGM. Znaków zastępczych należy używać tylko w ostatnim członie nazwy ścieżki; wpisanie na przykład /moj_katalog*/nazwa_pliku spowoduje wyświetlenie komunikatu o błędzie. Aby użyć funkcji **Przeglądaj** do wyświetlenia listy zawartości biblioteki lub katalogu, należy użyć znaków zastępczych w nazwie ścieżki, a następnie kliknąć przycisk **Przeglądaj**.
7. Wybierz opcje przetwarzania, których chcesz użyć do podpisywania wybranych obiektów, i kliknij **Kontynuuj**.

   Uwaga: Wybranie opcji oczekiwania na wyniki zadania spowoduje wyświetlenie pliku wynikowego bezpośrednio w przeglądarce. Wyniki bieżącego zadania zostaną dopisane na końcu pliku wyników. W rezultacie plik ten może zawierać oprócz wyników bieżącego zadania także wyniki poprzednich zadań. Aby zaznaczyć, które wiersze pliku odnoszą się do bieżącego zadania, można użyć pola daty. Pole to ma format RRRRMMDD. Pierwszym polem w pliku może być albo ID komunikatu (jeśli podczas przetwarzania obiektów wystąpił błąd) albo pole daty (określające datę przetwarzania zadania).
8. Podaj pełną ścieżkę i nazwę pliku do zapisywania wyników zadania dla operacji podpisywania obiektów i kliknij **Kontynuuj**. Można także wpisać ścieżkę do katalogu i kliknąć **Przeglądaj**, aby wyświetlić zawartość tego katalogu i wybrać plik do zapisywania wyników zadania. Wyświetli się komunikat informujący, że zostało uruchomione podpisanie obiektów. Aby wyświetlić wyniki zadania, znajdź zadanie **QOBJSGNBAT** w protokole zadania.

Aby sprawdzić, czy użytkownik lub inne osoby mogą weryfikować podpisy, należy najpierw wyeksportować niezbędne certyfikaty do pliku, a plik certyfikatów przesłać do systemu B. Przed przesłaniem podpisanych obiektów programów do systemu B należy także zakończyć wszystkie zadania konfiguracyjne związane z weryfikowaniem podpisów w tym systemie. Przed zweryfikowaniem podpisów odtworzonych obiektów w systemie B należy zakończyć konfigurowanie weryfikowania podpisów.

Czynność 6: eksportowanie certyfikatów umożliwiających weryfikowanie podpisów w systemie B

Podpisywanie obiektów w celu ochrony integralności ich zawartości ma sens tylko wtedy, gdy istnieją metody weryfikowania autentyczności podpisu. Aby zweryfikować podpisy obiektów w tym samym systemie, który podpisuje obiekty (system A), należy użyć programu DCM w celu utworzenia bazy certyfikatów *SIGNATUREVERIFICATION. Musi ona zawierać zarówno kopię certyfikatu podpisującego obiekt, jak i kopię certyfikatu ośrodka CA, który wystawił certyfikat podpisujący.

Aby inni użytkownicy mogli weryfikować podpis, należy im dostarczyć kopię certyfikatu podpisującego obiekt. Jeśli do wystawienia certyfikatu korzysta się z lokalnego ośrodka CA, trzeba im zapewnić także kopię certyfikatu lokalnego ośrodka CA.

Aby za pomocą programu DCM weryfikować podpisy w tym samym systemie, który podpisuje obiekty (w tym scenariuszu jest to system A), wykonaj następujące czynności:

1. W oknie nawigacji wybierz **Tworzenie nowej bazy certyfikatów**, a następnie, jako bazę certyfikatów do utworzenia wybierz ***SIGNATUREVERIFICATION**.
2. Wybierz **Tak**, aby skopiować do nowej bazy certyfikatów istniejące certyfikaty podpisujące obiekty jako certyfikaty weryfikujące podpisy.
3. Podaj hasło dla nowej bazy certyfikatów i kliknij **Kontynuuj**, aby ją utworzyć. Od tego momentu za pomocą programu DCM możesz weryfikować podpisy obiektów na tym samym systemie, którego używasz do podpisywania obiektów.

Aby użyć programu DCM do eksportowania kopii certyfikatu lokalnego CA i kopii certyfikatu podpisującego obiekty jako certyfikatu weryfikującego podpisy w celu weryfikacji podpisów obiektów na innych systemach (system B), wykonaj następujące czynności:

1. W ramce nawigacji wybierz **Zarządzanie certyfikatami**, a następnie wybierz zadanie **Eksport certyfikatu**.
2. Wybierz opcję **Ośrodek certyfikacji (CA)** i kliknij przycisk **Kontynuuj**, aby wyświetlić listę certyfikatów CA, które można wyeksportować.
3. Wybierz z listy uprzednio utworzony certyfikat lokalnego CA i kliknij **Eksportuj**.
4. Określ **Plik** jako miejsce docelowe eksportowania i kliknij przycisk **Kontynuuj**.
5. Podaj pełną ścieżkę i nazwę dla pliku eksportowanego certyfikatu lokalnego CA i kliknij **Kontynuuj**, aby wyeksportować certyfikat.
6. Kliknij **OK**, aby opuścić stronę potwierdzenia eksportu. Możesz już eksportować kopię certyfikatu podpisującego obiekty.
7. Ponownie wybierz zadanie **Eksportuj certyfikat**.
8. Wybierz **Podpisywanie obiektów**, aby wyświetlić listę certyfikatów podpisujących obiekty, które można eksportować.
9. Wybierz z listy odpowiedni certyfikat podpisujący obiekty i kliknij opcję **Eksportuj**.
10. Wybierz **Plik, jako certyfikat do weryfikowania podpisów** jako miejsce docelowe i kliknij **Kontynuuj**.
11. Podaj pełną ścieżkę i nazwę dla pliku eksportowanego certyfikatu do weryfikowania podpisów i kliknij **Kontynuuj**, aby wyeksportować certyfikat.

Od tego momentu można przesyłać te pliki do systemów końcowych, na których mają być weryfikowane podpisy utworzone za pomocą certyfikatu.

Czynność 7: przesłanie plików certyfikatów na publiczny serwer przedsiębiorstwa (system B)

Przed skonfigurowaniem systemów w celu weryfikowania podpisanych obiektów należy przesłać pliki certyfikatów utworzone w systemie A do systemu B, publicznego serwera WWW przedsiębiorstwa w ramach tego scenariusza. Do przesłania plików certyfikatów można użyć kilku metod, na przykład skorzystać z protokołu FTP lub z

rozpowszechniania pakietów w programie Centrum Zarządzania.

Czynność 8: zadania z zakresu weryfikowania podpisu: tworzenie bazy certyfikatów *SIGNATUREVERIFICATION

Aby weryfikować podpisy obiektów w systemie B (na publicznym serwerze WWW przedsiębiorstwa), system B musi mieć kopię odpowiedniego certyfikatu weryfikującego podpisy w bazie certyfikatów *SIGNATUREVERIFICATION. Ponieważ do podpisywania obiektów korzysta się z certyfikatu wystawionego przez lokalny ośrodek CA, baza certyfikatów musi zawierać także kopię jego certyfikatu.

Aby utworzyć bazę certyfikatów *SIGNATUREVERIFICATION, wykonaj następujące czynności:

1. Uruchom program DCM. Więcej informacji zawiera temat Uruchamianie programu DCM.
2. W ramce nawigacyjnej programu Digital Certificate Manager wybierz opcję **Tworzenie nowej bazy certyfikatów**, a następnie wybierz ***SIGNATUREVERIFICATION** jako nową bazę certyfikatów.

   Uwaga: W przypadku wątpliwości dotyczących określonego formularza podczas korzystania z programu DCM należy kliknąć przycisk znaku zapytania (**?**) u góry strony, aby uzyskać dostęp do systemu pomocy.
3. Podaj hasło dla nowej bazy certyfikatów i kliknij **Kontynuuj**, aby ją utworzyć. Możesz teraz importować certyfikaty do bazy i korzystać z nich do weryfikowania podpisów obiektów.

Czynność 9: zadania z zakresu weryfikowania podpisu: importowanie certyfikatów

Aby weryfikować podpis na obiekcie, baza *SIGNATUREVERIFICATION musi zawierać kopię certyfikatu do weryfikowania podpisów. Jeśli certyfikat podpisujący jest prywatny, to w bazie certyfikatów musi znaleźć się również kopia certyfikatu lokalnego ośrodka CA, który go wystawił. W opisywanym scenariuszu obydwa certyfikaty zostały wyeksportowane do pliku, który został przesłany do każdego systemu końcowego.

Wykonaj następujące czynności, aby zaimportować te certyfikaty do bazy *SIGNATUREVERIFICATION: Od tego momentu można już używać programu DCM w systemie B w celu weryfikowania podpisów obiektów utworzonych za pomocą odpowiedniego certyfikatu podpisującego w systemie A.

1. W oknie nawigacji programu DCM kliknij **Wybór ośrodka certyfikacji** i wybierz ***SIGNATUREVERIFICATION** jako bazę certyfikatów do otwarcia.
2. Po wyświetleniu strony Baza certyfikatów i hasło wpisz hasło określone dla bazy certyfikatów przy jej tworzeniu i kliknij przycisk **Kontynuuj**.
3. Po odświeżeniu widoku ramki nawigacji wybierz opcję **Zarządzanie certyfikatami**, aby wyświetlić listę zadań.
4. Z listy zadań wybierz **Import certyfikatu**.
5. Jako typ certyfikatu wybierz **Ośrodek certyfikacji** i kliknij **Kontynuuj**.

   Uwaga: Przed zaimportowaniem prywatnego certyfikatu do weryfikowania podpisów należy zaimportować certyfikat lokalnego ośrodka CA, inaczej proces importu certyfikatu do weryfikowania podpisów nie powiedzie się.
6. Podaj pełną ścieżkę i nazwę dla pliku certyfikatu ośrodka CA i kliknij **Kontynuuj**. Pojawi się komunikat potwierdzający pomyślne zakończenie importu lub informacja o błędzie, jeśli proces nie powiódł się.
7. Ponownie wybierz zadanie **Import certyfikatu**.
8. Jako typ certyfikatu wybierz **Sprawdzania podpisu** i kliknij **Kontynuuj**.
9. Podaj pełną ścieżkę i nazwę dla pliku certyfikatu do weryfikowania podpisów i kliknij **Kontynuuj**. Pojawi się komunikat potwierdzający pomyślne zakończenie importu lub informacja o błędzie, jeśli proces nie powiódł się.

Czynność 10: zadania z zakresu weryfikowania podpisu: weryfikowanie podpisów obiektów programów

Aby użyć programu do weryfikowania podpisów na przesłanych obiektach programów, wykonaj następujące czynności:

1. W ramce nawigacyjnej kliknij opcję **Wybór bazy certyfikatów**, a następnie wybierz ***SIGNATUREVERIFICATION** jako bazę certyfikatów, która ma zostać otworzona.
2. Wpisz hasło do bazy certyfikatów *SIGNATUREVERIFICATION i kliknij **Kontynuuj**.
3. Po odświeżeniu widoku ramki nawigacji wybierz **Zarządzanie obiektami do podpisywania**, aby wyświetlić listę zadań.
4. Z listy zadań wybierz **Sprawdzanie podpisu obiektu**, aby określić położenie obiektów, dla których chcesz weryfikować podpisy.
5. W wyświetlone pola wpisz pełną ścieżkę i nazwę pliku obiektu lub katalogu obiektów, których podpisy chcesz zweryfikować, i kliknij **Kontynuuj**. Można również wpisać położenie katalogu i kliknąć **Przeglądaj**, aby przejrzeć zawartość katalogu i wybrać obiekty do weryfikacji podpisu.

   Uwaga: Do określenia części obiektów katalogu, które mają zostać zweryfikowane, można także użyć znaków zastępczych. Te znaki to gwiazdka (*****), która zastępuje *dowolny ciąg znaków*, i znak zapytania (**?**), który zastępuje *dowolny pojedynczy znak*. Na przykład, aby podpisać wszystkie obiekty w określonym katalogu, można wpisać /mydirectory/* ; aby podpisać wszystkie programy w określonej bibliotece, można wpisać /QSYS.LIB/QGPL.LIB/*.PGM. Znaków zastępczych należy używać tylko w ostatnim członie nazwy ścieżki; wpisanie na przykład /moj_katalog*/nazwa_pliku spowoduje wyświetlenie komunikatu o błędzie. Aby użyć funkcji Przeglądaj do wyświetlenia listy zawartości biblioteki lub katalogu, należy użyć znaków zastępczych w nazwie ścieżki, a następnie kliknąć przycisk **Przeglądaj**.
6. Wybierz opcje przetwarzania, których chcesz użyć do weryfikowania podpisów wybranych obiektów, i kliknij **Kontynuuj**.

   Uwaga: Wybranie opcji oczekiwania na wyniki zadania spowoduje wyświetlenie pliku wynikowego bezpośrednio w przeglądarce. Wyniki bieżącego zadania zostaną dopisane na końcu pliku wyników. W rezultacie plik ten może zawierać oprócz wyników bieżącego zadania także wyniki poprzednich zadań. Aby zaznaczyć, które wiersze pliku odnoszą się do bieżącego zadania, można użyć pola daty. Pole to ma format RRRRMMDD. Pierwszym polem w pliku może być albo ID komunikatu (jeśli podczas przetwarzania obiektów wystąpił błąd) albo pole daty (określające datę przetwarzania zadania).
7. Podaj pełną ścieżkę i nazwę pliku do zapisywania wyników zadania dla operacji weryfikacji podpisów obiektów i kliknij **Kontynuuj**. Można także wpisać ścieżkę do katalogu i kliknąć **Przeglądaj**, aby wyświetlić zawartość tego katalogu i wybrać plik do zapisywania wyników zadania. Wyświetli się komunikat informujący, że zostało wprowadzone zadanie w celu weryfikacji podpisów obiektów. Aby wyświetlić wyniki zadania, znajdź zadanie **QOBJSGNBAT** w protokole zadania.

Scenariusz: podpisywanie obiektów i weryfikowanie podpisów obiektów za pomocą funkcji API

W scenariuszu przedstawiono przedsiębiorstwo projektujące aplikacje, które chce sprzedawane przez siebie produkty podpisywać programowo, aby klienci podczas instalacji byli pewni, że aplikacje pochodzą od producenta i aby mieli możliwość wykrycia w nich zmian wprowadzonych przez osoby bez uprawnień. Sposób użycia funkcji API Sign Object i Add Verifier systemu i5/OS do podpisywania obiektów i aktywowania weryfikacji podpisów został opisany w oparciu o potrzeby przedsiębiorstwa i jego cele związane z bezpieczeństwem.

Sytuacja

Przedsiębiorstwo (MojaFirma) jest partnerem handlowym rozwijającym aplikacje dla klientów. Jako twórca oprogramowania dla przedsiębiorstwa jesteś odpowiedzialny za tworzenie pakietów rozpowszechnianych wśród klientów. Do utworzenia pakietu aplikacji używasz pewnych programów. Klienci mogą zamówić dysk CD-ROM lub pobrać aplikację z serwisu WWW.

Na bieżąco zapoznajesz się z nowinkami technicznymi, szczególnie dotyczącymi ochrony. Dlatego wiesz, że klienci są żywotnie zainteresowani źródłem pochodzenia i zawartością otrzymywanych lub pobieranych programów. Zdarzają się przypadki, że klienci myślą, że otrzymali lub pobrali produkt z zaufanego źródła, potem jednak okazuje się, że nie jest

to prawdziwe miejsce pochodzenia tego produktu. Czasami wynika to z faktu, że klienci instalują inny program, niż oczekiwali. Czasami zaś program okazuje się być innym programem lub został zamieniony i powoduje uszkodzenie systemu.

Tego rodzaju problemy nie są co prawda napotykane często przez klientów, użytkownik pragnie jednak zapewnić klientów, że otrzymywane przez nich aplikacje pochodzą naprawdę z firmy użytkownika. Chcesz także dostarczyć klientom narzędzia do sprawdzenia integralności oraz autentyczności pochodzenia instalowanych aplikacji.

Po analizie możliwych rozwiązań użytkownik zdecydował się na wykorzystanie możliwości podpisywania obiektów oferowanych przez system i5/OS w celu osiągnięcia wyznaczonych celów w zakresie ochrony. Cyfrowe podpisywanie aplikacji umożliwia klientom sprawdzenie, czy dane przedsiębiorstwo jest prawowitym źródłem aplikacji, które otrzymali lub pobrali. Ponieważ tworzenie pakietów aplikacji odbywa się programowo, zdecydowałeś się użyć funkcji API, aby w prosty sposób dołączyć podpisywanie obiektów do istniejącego procesu tworzenia pakietów. Ponadto podjąłeś decyzję o użyciu do podpisywania obiektów certyfikatu publicznego, aby proces weryfikowania podpisu był dla klientów instalujących produkt całkowicie przezroczysty.

Do tworzonego pakietu dołączasz kopię certyfikatu cyfrowego użytego do podpisania obiektu. Po otrzymaniu pakietu klient skorzysta z klucza publicznego certyfikatu do zweryfikowania podpisu na aplikacji. Dzięki temu procesowi klient może określić i sprawdzić źródło aplikacji oraz zyskać pewność, że zawartość aplikacji nie uległa zmianie od czasu jej podpisania.

Przykład ten jest praktycznym wprowadzeniem do programowego podpisywania obiektów, takich jak opracowywane aplikacje czy pakiety przeznaczone dla innych osób.

Zalety scenariusza

Scenariusz ten ma następujące zalety:

- Korzystanie z funkcji API do programowego tworzenia pakietów i podpisywania obiektów pozwala na redukcję czasu spędzonego na implementowaniu tego środka ochrony.
- Korzystanie z funkcji API do podpisywania obiektów podczas tworzenia pakietów zmniejsza ilość koniecznych działań, gdyż proces podpisywania jest częścią procesu tworzenia pakietów.
- Podpisywanie pakietu obiektów umożliwia łatwe określenie, czy obiekty zostały zmienione od czasu podpisania. Pozwala zmniejszyć nakłady na przyszłe rozwiązywanie i śledzenie problemów pojawiających się w aplikacjach działających u klientów.
- Wykorzystanie do podpisania obiektów certyfikatu powszechnie znanego publicznego ośrodka CA pozwala na zastosowanie w części programu obsługi wyjścia programu instalacyjnego produktu funkcji API Add Verifier. Wykorzystanie tej funkcji pozwala automatycznie dodać do systemu klienta publiczny certyfikat użyty do podpisania aplikacji. To zaś daje pewność, że weryfikowanie podpisu jest przezroczyste dla klienta.

Cele

W scenariuszu zakładamy, że przedsiębiorstwo MojaFirma chce programowo podpisywać tworzone pakiety aplikacji rozpowszechniane do klientów. Jako programista w przedsiębiorstwie MojaFirma programowo tworzysz pakiety aplikacji, które następnie są rozpowszechniane do klientów. Dlatego użytkownik pragnie użyć funkcji API systemu do podpisywania aplikacji w taki sposób, aby system klienta mógł programowo weryfikować podpis podczas instalowania produktu.

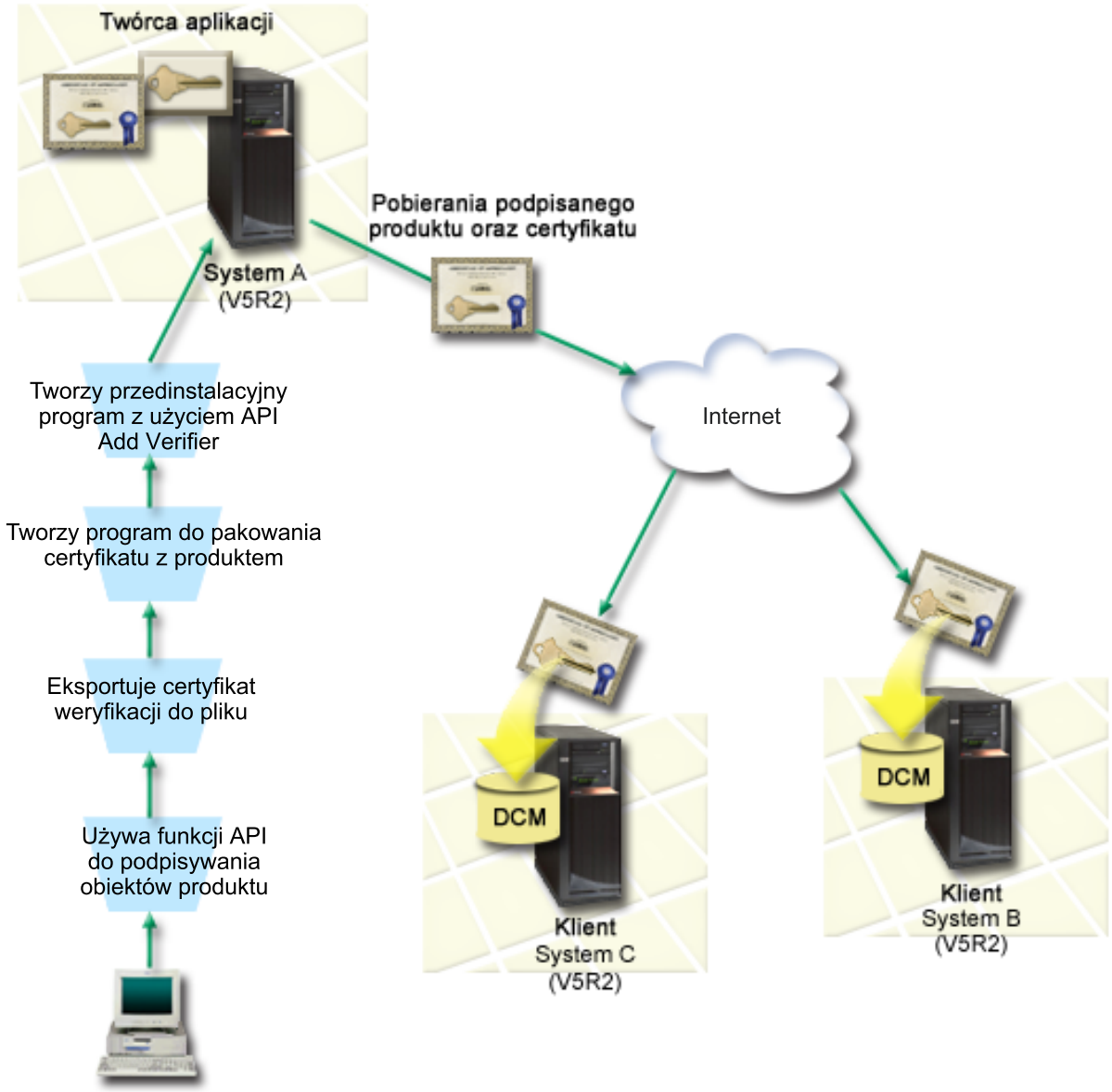
Główne założenia scenariusza są następujące:

- Dostawca aplikacji musi mieć możliwość podpisywania obiektów za pomocą funkcji API Sign Object w trakcie opracowanego już procesu tworzenia pakietu aplikacji.
- Aplikacje muszą być podpisane certyfikatem publicznym, aby proces weryfikowania podpisu, następujący w trakcie instalacji aplikacji, był całkowicie przezroczysty dla klienta.

- Firma musi mieć możliwość używania funkcji API systemu do programowego dodawania odpowiedniego certyfikatu weryfikującego podpis do bazy certyfikatów *SIGNATUREVERIFICATION w systemie klienta. Firma musi mieć możliwość programowego tworzenia tej bazy certyfikatów w systemie klienta jako część procesu instalowania produktu, jeśli baza ta jeszcze nie istnieje.
- Po zainstalowaniu produktu klienci muszą mieć możliwość prostego weryfikowania podpisów cyfrowych na aplikacji. Jest to niezbędne, aby ustalić źródło i autentyczność podpisanej aplikacji oraz określić, czy od czasu podpisania aplikacja nie została zmieniona.

Informacje szczegółowe

Poniższy rysunek przedstawia proces podpisywania obiektu i weryfikowania podpisu według scenariusza:



Rysunek ilustruje następujące punkty scenariusza:

System centralny A

- System A to serwer System i, na którym działa system operacyjny OS/400 Wersja 5, Wydanie 2 (V5R2).

- System A uruchamia program tworzący pakiety aplikacji opracowanej przez programistę.
- W systemie A zainstalowany jest produkt Cryptographic Access Provider 128-bit for System i (5722–AC3).
- W systemie A zainstalowano i skonfigurowano produkty Digital Certificate Manager (opcja 34) oraz IBM HTTP Server (5722–DG1).
- System A jest podstawowym systemem podpisującym obiekty dla aplikacji w przedsiębiorstwie. Podpisywanie obiektów produktów przeznaczonych do dystrybucji wśród klientów przebiega w systemie A w następujący sposób:
  1. Aplikacja jest podpisywana za pomocą funkcji API.
  2. Certyfikat do weryfikowania podpisów jest eksportowany do pliku za pomocą programu DCM, aby klienci mogli weryfikować podpisane obiekty.
  3. Powstaje program służący do dodawania certyfikatu weryfikującego do podpisywanej aplikacji.
  4. Powstaje program przedinstalacyjny obsługi wyjścia korzystający z funkcji API Add Verifier. Ta funkcja API zezwala procesowi instalowania produktów na programowe dodawanie certyfikatu weryfikującego do bazy certyfikatów *SIGNATUREVERIFICATION w systemie klienta (systemy B i C).

Systemy B i C klienta

- System B to serwer System i, na którym działa system operacyjny OS/400 Wersja 5, Wydanie 2 (V5R2) lub późniejsze wydanie systemu i5/OS.
- System C to serwer System i, na którym działa system operacyjny OS/400 Wersja 5, Wydanie 2 (V5R2) lub późniejsze wydanie systemu i5/OS.
- W systemach B i C zainstalowano i skonfigurowano produkty Digital Certificate Manager (opcja 34) oraz IBM HTTP Server (5722–DG1).
- Systemy B i C kupują i pobierają aplikację z serwisu WWW przedsiębiorstwa rozwijającego aplikacje (które jest właścicielem systemu A).
- Systemy B i C uzyskują kopie certyfikatu weryfikowania podpisów przedsiębiorstwa MojaFirma, gdy proces instalowania aplikacji przedsiębiorstwa MojaFirma tworzy bazę certyfikatów *SIGNATUREVERIFICATION w każdym z systemów klienta.

Wymagania wstępne i założenia

Scenariusz zależy od spełnienia następujących założeń i wymagań wstępnych:

1. Wszystkie systemy spełniają wymagania w zakresie instalowania i używania programu Digital Certificate Manager (DCM).

   Uwaga: Spełnienie przez klientów (w tym scenariuszu przez systemy B i C) wymagań wstępnych związanych z instalowaniem i korzystaniem z programu DCM jest wymogiem opcjonalnym. Jednak funkcja API Add Verifier podczas procesu instalacji produktu, jeśli jest to potrzebne, tworzy bazę certyfikatów *SIGNATUREVERIFICATION z hasłem domyślnym. Aby zaś lepiej chronić bazę certyfikatów przez dostępem bez uprawnień i móc zmienić to hasło, klienci muszą korzystać z programu DCM.
2. W żadnym systemie nie był wcześniej konfigurowany ani używany program DCM.
3. We wszystkich systemach zainstalowano najnowszą wersję programu licencjonowanego Cryptographic Access Provider 128-bit (5722-AC3).
4. Dla wartości systemowej Weryfikowanie podpisów obiektów podczas odtwarzania (VeriFY OBJect signatures during ReSTore - QVFYOBJRST) określono domyślną wartość 3 we wszystkich systemach objętych scenariuszem i nie została ona zmieniona. Ustawienie domyślne daje gwarancję, że system będzie mógł zweryfikować podpisy po odtworzeniu podpisanych obiektów.
5. Aby administrator sieci systemu A mógł podpisywać obiekty, musi mieć uprawnienia specjalne profilu użytkownika *ALLOBJ albo jego profil użytkownika musi być uprawniony do korzystania z aplikacji podpisującej obiekty.
6. Administrator systemu lub inna osoba, która tworzy bazę certyfikatów w programie DCM, musi mieć uprawnienia specjalne profilu użytkownika *SECADM i *ALLOBJ.

7. W celu weryfikowania podpisów obiektów administratorzy systemów lub inni użytkownicy wszystkich pozostałych systemów muszą mieć uprawnienia specjalne profilu użytkownika *AUDIT.

Czynności konfiguracyjne

Informacje dotyczące podpisywania obiektów według opisu przedstawionego w ramach tego scenariusza zamieszczono poniżej w szczegółach do scenariusza. Zawierają one listę czynności umożliwiającą wykonanie każdego z następujących zadań w systemie A:

1. Spełnienie wszystkich wymagań wstępnych koniecznych do zainstalowania i skonfigurowania wszystkich potrzebnych produktów serwera System i.
2. Używanie programu DCM do utworzenia żądania certyfikatu w celu uzyskania certyfikatu podpisującego obiekty od powszechnie znanego, publicznego ośrodka certyfikacji (CA).
3. Używanie programu DCM w celu utworzenia definicji aplikacji podpisującej obiekty.
4. Używanie programu DCM w celu importowania podpisanego certyfikatu do podpisywania obiektów i przypisania go definicji aplikacji podpisującej obiekty.
5. Używanie programu DCM w celu eksportowania certyfikatu podpisującego obiekty jako certyfikatu służącego do weryfikowania podpisów, aby klienci mogli go użyć do weryfikowania podpisów w obiektach aplikacji.
6. Aktualizowanie programu tworzenia pakietu aplikacji w taki sposób, aby do podpisywania aplikacji była używana funkcja API Sign Object.
7. Utworzenie przedinstalacyjnego programu obsługi wyjścia używającego funkcji API Add Verifier w ramach procesu tworzenia pakietu aplikacji. Program obsługi wyjścia umożliwia utworzenie podczas instalowania produktu bazy certyfikatów *SIGNATUREVERIFICATION i dodanie niezbędnego certyfikatu do weryfikowania podpisów w systemie klienta.
8. Korzystanie przez klientów z programu DCM w celu zresetowania domyślnego hasła bazy certyfikatów *SIGNATUREVERIFICATION w systemie.

Informacje pokrewne

Digital Certificate Manager (DCM)

Szczegóły scenariusza: podpisywanie obiektów i weryfikowanie podpisów obiektów za pomocą funkcji API

Aby zgodnie ze scenariuszem użyć funkcji API systemu i5/OS do podpisywania obiektów, należy wykonać poniższe czynności.

Czynność 1: spełnienie wszystkich wymagań wstępnych

Przed wykonaniem określonych zadań konfiguracyjnych związanych z implementacją tego scenariusza należy w całości spełnić wymagania wstępne w zakresie instalowania i konfigurowania wszystkich potrzebnych produktów serwera System i.

Czynność 2: zastosowanie programu DCM w celu pobrania certyfikatu z powszechnie znanego, publicznego ośrodka CA

Scenariusz zakłada, że wcześniej nie używano z programu Digital Certificate Manager do tworzenia i zarządzania certyfikatami. Dlatego jako część procesu tworzenia własnego certyfikatu podpisującego obiekty należy utworzyć bazę certyfikatów *OBJECTSIGNING. Podczas tworzenia bazy certyfikatów zrealizowane zostaną zadania potrzebne do tworzenia i zarządzania certyfikatami podpisującymi obiekty. Aby uzyskać od powszechnie znanego publicznego ośrodka CA certyfikat, użyj programu DCM do utworzenia informacji identyfikujących i pary kluczy publiczny-prywatny dla certyfikatu i wysłania tych informacji do ośrodka CA.

Aby utworzyć wniosek o certyfikat, który należy dostarczyć do powszechnie znanego publicznego ośrodka CA w celu otrzymania certyfikatu podpisującego obiekty, wykonaj następujące czynności:

1. Uruchom program DCM. Więcej informacji zawiera temat Uruchamianie programu DCM.

2. W ramce nawigacji programu DCM wybierz **Tworzenie nowej bazy certyfikatów**, aby rozpocząć procedurę i wypełnić szereg formularzy. Formularze te prowadzą przez proces tworzenia bazy certyfikatów i certyfikatu, którego można będzie używać do podpisywania obiektów.

   Uwaga: W przypadku pojawienia się pytań dotyczących określonego formularza w tym zadaniu należy kliknąć przycisk znaku zapytania (**?**) u góry strony, aby uzyskać dostęp do systemu pomocy.
3. Wybierz ***OBJECTSIGNING** jako bazę certyfikatów, która ma zostać utworzona, a następnie kliknij przycisk **Kontynuuj**.
4. Wybierz **Tak**, aby w ramach tworzenia bazy certyfikatów *OBJECTSIGNING utworzyć certyfikat, i kliknij **Kontynuuj**.
5. Jako ośrodek podpisujący nowy certyfikat wybierz **VeriSign lub inny internetowy ośrodek certyfikacji** i kliknij **Kontynuuj**, aby wyświetlić formularz pozwalający podać informacje identyfikujące dla nowego certyfikatu.
6. Wypełnij formularz i kliknij **Kontynuuj**, aby wyświetlić stronę potwierdzenia. Na stronie tej wyświetlane są dane do wniosku, który należy dostarczyć do ośrodka certyfikacji wystawiającego certyfikat. Dane Certificate Signing Request (CSR) zawierają klucz publiczny i inne informacje podane do certyfikatu.
7. Uważnie skopiuj dane CSR i wklej je do formularza wniosku o certyfikat lub do osobnego pliku wymaganego przez publiczny ośrodek przy występowaniu o certyfikat. Należy użyć wszystkich danych CSR, w tym również wierszy Początek wniosku o nowy certyfikat i Koniec wniosku o nowy certyfikat. Po zamknięciu tej strony dane zostaną utracone i nie będzie można ich odtworzyć.
8. Formularz wniosku lub plik należy wysłać do wybranego ośrodka certyfikacji, który ma wystawić i podpisać certyfikat.
9. Przed przejściem do następnego zadania w tym scenariuszu zaczekaj aż ośrodek CA odeśle podpisany, wypełniony certyfikat.

Czynność 3: tworzenie definicji aplikacji podpisującej obiekty

Po wysłaniu wniosku o certyfikat do powszechnie znanego, publicznego ośrodka CA, można użyć programu DCM do utworzenia definicji aplikacji podpisującej obiekty, która będzie służyć do podpisywania obiektów. Definicja aplikacji nie musi odnosić się do istniejącej aplikacji. Tworzona definicja aplikacji powinna opisywać typ lub grupę obiektów, które zamierzasz podpisywać. Definicja jest niezbędna, żeby można było powiązać identyfikator aplikacji z certyfikatem i umożliwić proces podpisywania.

Aby utworzyć definicję aplikacji podpisującej obiekty za pomocą programu DCM, wykonaj następujące czynności:

1. W ramce nawigacyjnej kliknij opcję **Wybór bazy certyfikatów** i wybierz ***OBJECTSIGNING** jako bazę certyfikatów, która ma zostać otworzona.
2. Po wyświetleniu strony Baza certyfikatów i hasło wpisz hasło określone dla bazy certyfikatów przy jej tworzeniu i kliknij przycisk **Kontynuuj**.
3. W ramce nawigacji wybierz **Zarządzanie aplikacjami**, aby wyświetlić listę zadań.
4. Z listy zadań wybierz **Dodaj aplikację**, aby wyświetlić formularz definiowania aplikacji.
5. Wypełnij formularz i kliknij **Dodaj**.

Po otrzymaniu od ośrodka CA podpisanego certyfikatu można przypisać certyfikat do utworzonej aplikacji.

Czynność 4: importowanie podpisanego certyfikatu publicznego i przypisanie go aplikacji podpisującej obiekty

Aby zaimportować certyfikat, przypisać go do aplikacji i włączyć podpisywanie obiektów, wykonaj następujące czynności:

1. Uruchom program DCM. Więcej informacji zawiera temat Uruchamianie programu DCM.
2. W ramce nawigacyjnej kliknij opcję **Wybór bazy certyfikatów** i wybierz ***OBJECTSIGNING** jako bazę certyfikatów, która ma zostać otworzona.

3. Po wyświetleniu strony Baza certyfikatów i hasło wpisz hasło określone dla bazy certyfikatów przy jej tworzeniu i kliknij przycisk **Kontynuuj**.
4. Po odświeżeniu widoku ramki nawigacji wybierz opcję **Zarządzanie certyfikatami**, aby wyświetlić listę zadań.
5. Z listy zadań wybierz **Import certyfikatu**, aby rozpocząć proces importowania podpisanego certyfikatu do bazy certyfikatów.

   Uwaga: W przypadku pojawienia się pytań dotyczących określonego formularza w tym zadaniu należy kliknąć przycisk znaku zapytania (**?**) u góry strony, aby uzyskać dostęp do systemu pomocy.
6. Wybierz **Przypisanie certyfikatu** z listy zadań **Zarządzanie certyfikatami**, aby wyświetlić listę certyfikatów bieżącej bazy certyfikatów.
7. Wybierz certyfikat z listy i kliknij **Przypisz do aplikacji**, aby wyświetlić listę definicji aplikacji bieżącej bazy certyfikatów.
8. Wybierz swoją aplikację z listy i kliknij **Kontynuuj**. Pojawi się strona z komunikatem potwierdzającym wybór przypisania, albo komunikat o błędzie, jeśli wystąpi jakiś problem.

Po zakończeniu tych czynności użytkownik może korzystać z podpisywania aplikacji i innych obiektów za pomocą funkcji API systemu i5/OS. Aby jednak mieć pewność, że użytkownik i inne osoby mogą weryfikować podpisy, należy wyeksportować niezbędne certyfikaty do pliku i przesłać je do systemu, w którym instalowane są podpisane aplikacje. Systemy klienta muszą być przygotowane na używanie certyfikatu, aby zweryfikować podpis na aplikacji podczas jej instalowania. Do konfigurowania weryfikacji podpisów u klientów możesz użyć funkcji API Add Verifier jako części swojego programu instalacyjnego aplikacji. Można na przykład utworzyć przedinstalacyjny program obsługi wyjścia wywołujący funkcję API Add Verifier w celu skonfigurowania systemu klienta.

Czynność 5: eksportowanie certyfikatów w celu umożliwienia weryfikowania podpisów w innych systemach

Podpisywanie obiektów ma sens tylko wtedy, gdy istnieją metody weryfikowania autentyczności podpisu i określania czy do podpisanego obiektu zostały wprowadzone jakieś zmiany. Aby zweryfikować podpisy obiektów na tym samym systemie, który podpisuje obiekty, należy użyć programu DCM do utworzenia bazy certyfikatów *SIGNATUREVERIFICATION. Musi ona zawierać zarówno kopię certyfikatu podpisującego obiekt, jak i kopię certyfikatu ośrodka CA, który wystawił certyfikat podpisujący.

Aby inni użytkownicy mogli weryfikować podpis, należy im dostarczyć kopię certyfikatu podpisującego obiekt. Jeśli do wystawienia certyfikatu korzysta się z lokalnego ośrodka CA, trzeba im zapewnić także kopię certyfikatu lokalnego ośrodka CA.

Aby za pomocą programu DCM weryfikować podpisy w tym samym systemie, który podpisuje obiekty (w tym scenariuszu jest to system A), wykonaj następujące czynności:

1. W oknie nawigacji wybierz **Tworzenie nowej bazy certyfikatów**, a następnie, jako bazę certyfikatów do utworzenia wybierz ***SIGNATUREVERIFICATION**.
2. Wybierz **Tak**, aby skopiować do nowej bazy certyfikatów istniejące certyfikaty podpisujące obiekty jako certyfikaty weryfikujące podpisy.
3. Podaj hasło dla nowej bazy certyfikatów i kliknij **Kontynuuj**, aby ją utworzyć. Od tego momentu za pomocą programu DCM możesz weryfikować podpisy obiektów na tym samym systemie, którego używasz do podpisywania obiektów.

Aby użyć programu DCM do eksportowania kopii certyfikatu podpisującego obiekt jako certyfikatu weryfikującego podpis, tak aby inni użytkownicy mogli weryfikować podpisy obiektów, wykonaj następujące czynności:

1. W ramce nawigacji wybierz **Zarządzanie certyfikatami**, a następnie wybierz zadanie **Eksport certyfikatu**.
2. Wybierz **Podpisywanie obiektów**, aby wyświetlić listę certyfikatów podpisujących obiekty, które można eksportować.
3. Wybierz z listy odpowiedni certyfikat podpisujący obiekty i kliknij opcję **Eksportuj**.
4. Wybierz **Plik, jako certyfikat do weryfikowania podpisów** jako miejsce docelowe i kliknij **Kontynuuj**.

5. Podaj pełną ścieżkę i nazwę dla pliku eksportowanego certyfikatu do weryfikowania podpisów i kliknij **Kontynuuj**, aby wyeksportować certyfikat.

Możesz teraz dodawać ten plik do pakietów instalacyjnych aplikacji tworzonych dla produktu. Za pomocą funkcji API Add Verifier jako części programu instalacyjnego możesz dodać ten certyfikat do bazy certyfikatów *SIGNATUREVERIFICATION klienta. Jeśli baza certyfikatów jeszcze nie istnieje, funkcja ta ją utworzy. Następnie program instalacyjny produktu może zweryfikować podpis na obiektach aplikacyjnych podczas odtwarzania ich w systemach klienta.

Czynność 6: aktualizowanie programu tworzącego pakiet aplikacji, aby używał systemowych funkcji API do podpisywania aplikacji

Gdy masz już plik zawierający certyfikat do weryfikowania podpisów i chcesz go dodać do pakietu aplikacji, możesz skorzystać z funkcji API Sign Object i napisać nową lub zmienić istniejącą aplikację w taki sposób, aby podpisywała biblioteki produktu podczas tworzenia pakietu rozpowszechnianego wśród klientów.

Aby lepiej zrozumieć wykorzystanie funkcji API Sign Object jako części programu tworzącego pakiet aplikacji, przejrzyj przykład przedstawiony poniżej. Przykład ten jest fragmentem kodu programu napisanego w języku C, nie jest on pełnym programem do podpisywania i tworzenia pakietów, jest to raczej wycinek programu, w którym następuje wywołanie funkcji API Sign Object. Jeśli chcesz skorzystać z tego przykładu, dostosuj go do swoich potrzeb. Ze względów bezpieczeństwa firma IBM zaleca zindywidualizowanie przykładu i zmianę dostarczonych wartości domyślnych.

Uwaga: Korzystając z przykładów kodu, użytkownik wyraża zgodę na warunki zapisane w sekcji “Licencja na kod oraz Informacje dotyczące kodu” na stronie 47.

Dostosuj ten przykład do swoich potrzeb w celu wykorzystania funkcji API Sign Object jako części programu tworzącego pakiety aplikacji. Do programu należy dostarczyć dwa parametry: nazwę biblioteki do podpisania i nazwę identyfikatora aplikacji podpisującej obiekt; w identyfikatorze aplikacji rozróżnia się wielkie i małe litery, których nie rozróżnia się w nazwie biblioteki. Jeśli podpisywany obiekt składa się z kilku bibliotek, program może wywoływać ten fragment kodu wielokrotnie.

```
/* ------------------------------------------------------------- */
/*                                                               */
/* COPYRIGHT (C) IBM CORP. 2004, 2007                            */
/*                                                               */
/* Wykorzystanie funkcji API Sign Object do podpisywania bibliotek  */
/*                                                               */
/* Funkcja API podpisuje cyfrowo wszystkie obiekty w bibliotece     */
/*                                                               */
/*                                                               */
/*                                                               */
/* IBM udziela niewyłącznej licencji na prawa autorskie, stosowanej */
/* przy używaniu wszelkich przykładowych kodów programów, na        */
/* podstawie których można wygenerować podobne funkcje dostosowane  */
/* do indywidualnych wymagań. Cały kod przykładowy jest udostępniany*/
/* przez IBM jedynie do celów ilustracyjnych. Programy przykładowe  */
/* nie zostały gruntownie przetestowane. IBM nie może gwarantować   */
/* lub sugerować niezawodności, użyteczności i funkcjonalności      */
/* tych programów. Wszystkie zawarte tu programy są dostarczane     */
/* w stanie, w jakim się znajdują ("AS IS") bez udzielania          */
/* jakichkolwiek gwarancji. Nie udziela się domniemanych gwarancji  */
/* przydatności handlowej oraz przydatności do  określonego celu.   */
/*                                                               */
/*                                                               */
/*                                                               */
/* Parametry programu:                                           */
/*                                                               */
/*  char *   nazwa podpisywanej biblioteki                       */
/*  char *   nazwa identyfikatora aplikacji                      */
/*                                                               */
```

```
#include <qydosgno.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

int main (int argc, char *argv[])
{

    /* parametry:

       char * biblioteka obiektów do podpisania
       char * identyfikator aplikacji podpisującej

     */

    int         lib_length, applid_length, path_length, multiobj_length;
    Qus_EC_t    error_code;
    char        libname[11];
    char        path_name[256];

    Qydo_Multi_Objects_T * multi_objects = NULL;
    multiobj_length = 0;
    error_code.Bytes_Provided = 0;     /* wyjątki zwracane dla błędów       */


    /* ------------------------------------ */
    /* budowa nazwy ścieżki dla biblioteki    */
    /* ------------------------------------ */
    memset(libname, '\00', 11); /* inicjowanie nazwy biblioteki */
    for(lib_length = 0;
        ((*(argv[1] + lib_length) != ' ') &&
         (*(argv[1] + lib_length) != '\00'));
        lib_length++);
    memcpy(argv[1], libname, lib_length); /* wpisanie nazwy biblioteki */

    /* budowa parametru nazwa ścieżki do wywołania funkcji API */
    sprintf(path_name, "/QSYS.LIB/%s.LIB/*", libname);
    path_length = strlen(path_name);

    /* ---------------------------- */
    /* szukanie długości ID aplikacji*/
    /* ---------------------------- */
    for(applid_length = 0;
        ((*(argv[2] + applid_length) != ' ') &&
         (*(argv[2] + applid_length) != '\00'));
        applid_length++);

    /* ------------------------------- */
    /* podpisanie obiektów w bibliotece */
    /* ------------------------------- */
    QYDOSGNO  (path_name,             /* nazwa ścieżki do obiektu    */
               &path_length,          /* długość nazwy ścieżki       */
               "OBJN0100",       /* nazwa formatu                     */
               argv[2],               /* identyfikator aplikacji (ID)*/
               &applid_length,        /* długość ident. aplikacji    */
               "1",                   /* zastąpienie podwójnego podp.*/
               multi_objects,         /* jak obsługiwać wiele
                                         obiektów                */
               &multiobj_length,      /* długość używanej struktury
                                           zawiera wiele obiektów
                                      (0=bez str. zaw. wiele ob.) */
               &error_code);     /* kod błędu                        */
```

```
    return 0;

}
```

Czynność 7: tworzenie programu przedinstalacyjnego obsługi wyjścia korzystającego z funkcji API Add Verifier

Po zakończeniu procesu tworzenia programu do podpisywania aplikacji możesz korzystać z funkcji API Add Verifier jako części programu instalacyjnego do tworzenia rozpowszechnianego produktu końcowego. Możesz na przykład użyć funkcji API Add Verifier jako części programu przedinstalacyjnego obsługi wyjścia. Zyskasz pewność, że przed odtworzeniem podpisanych obiektów aplikacji certyfikat zostanie dodany do bazy certyfikatów. Umożliwi to programowi instalacyjnemu weryfikację podpisu na obiektach aplikacji podczas odtwarzania ich w systemie klienta.

Uwaga: Ze względów bezpieczeństwa ta funkcja API nie pozwoli na dodanie do bazy certyfikatów *SIGNATUREVERIFICATION certyfikatu ośrodka certyfikacji. W momencie dodawania certyfikatu ośrodka certyfikacji do bazy certyfikatów system zakłada, że ośrodek CA jest zaufanym źródłem certyfikatów. W konsekwencji system traktuje certyfikat wystawiony przez ośrodek CA jako pochodzący z zaufanego źródła. Dlatego nie można korzystać z funkcji API w celu utworzenia programu instalacyjnego obsługi wyjścia w celu dodania certyfikatu ośrodka certyfikacji do bazy certyfikatów. Aby dodać certyfikat ośrodka CA do bazy certyfikatów, należy użyć programu Digital Certificate Manager; takie rozwiązanie daje pewność, że sprawowana jest ręczna, dokładna kontrola nad ośrodkami CA, którym system ufa. Dzięki temu system nie może importować certyfikatów ze źródeł, których administrator świadomie nie określił jako zaufane.

Jeśli nie chcesz, aby ktokolwiek mógł dodać certyfikat weryfikacji do bazy certyfikatów *SIGNATUREVERIFICATION, korzystając z tej funkcji API, zablokuj tę funkcję w systemie. Aby zrealizować to zadanie, skorzystaj z systemowych narzędzi serwisowych (SST) umożliwiających ustawienie odrzucania zmian wprowadzonych w wartościach systemowych związanych z ochroną.

Aby lepiej zrozumieć wykorzystanie funkcji API Add Verifier jako części programu instalacyjnego aplikacji, przejrzyj przedstawiony poniżej przykład programu przedinstalacyjnego obsługi wyjścia. Przykład ten jest fragmentem kodu programu napisanego w języku C, nie jest on pełnym programem przedinstalacyjnym obsługi wyjścia, jest to raczej wycinek programu, w którym następuje wywołanie funkcji API Add Verifier. Jeśli chcesz skorzystać z tego przykładu, dostosuj go do swoich potrzeb. Ze względów bezpieczeństwa firma IBM zaleca zindywidualizowanie przykładu i zmianę dostarczonych wartości domyślnych.

Uwaga: Korzystając z przykładowego kodu użytkownik akceptuje warunki opisane w temacie “Licencja na kod oraz Informacje dotyczące kodu” na stronie 47.

Dostosuj ten fragment kodu do własnych potrzeb w celu wykorzystania funkcji API Add Verifier jako części programu przedinstalacyjnego obsługi wyjścia umożliwiającego dodanie niezbędnego certyfikatu weryfikującego podpisy w systemie klienta podczas instalowania produktu użytkownika.

```
/* ------------------------------------------------------------ */
/*                                                              */
/* COPYRIGHT (C) IBM CORP. 2004, 2007                           */
/*                                                              */
/* Wykorzystanie funkcji API Add Verifier do dodania certyfikatu    */
/* zintegrowanego systemu plików bazy certyfikatów                  */
/* *SIGNATUREVERIFICATION.                                      */
/*                                                              */
/*                                                              */
/* Jeśli baza certyfikatów nie istnieje, funkcja API utworzy ją.    */
/* Jeśli baza certyfikatów jest tworzona, otrzyma hasło domyślne,   */
/* które należy zmienić najszybciej jak to możliwe korzystając z    */
/* programu DCM. To ostrzeżenie należy przedstawić właścicielowi    */
/* systemu, który będzie korzystał z tego programu.                 */
/*                                                              */
/*                                                              */
```

```
/*                                                                  */
/* IBM udziela niewyłącznej licencji na prawa autorskie, stosowanej */
/* przy używaniu wszelkich przykładowych kodów programów, na        */
/* podstawie których można wygenerować podobne funkcje dostosowane  */
/* do indywidualnych wymagań. Cały kod przykładowy jest udostępniany*/
/* przez IBM jedynie do celów ilustracyjnych. Programy przykładowe  */
/* nie zostały gruntownie przetestowane. IBM nie może gwarantować   */
/* lub sugerować niezawodności, użyteczności i funkcjonalności      */
/* tych programów. Wszystkie zawarte tu programy są dostarczane     */
/* w stanie, w jakim się znajdują ("AS IS") bez udzielania          */
/* jakichkolwiek gwarancji. Nie udziela się domniemanych gwarancji  */
/* przydatności handlowej oraz przydatności do  określonego celu.   */
/*                                                                  */
/*                                                                  */
/*                                                                  */
/* Parametry programu:                                              */
/*                                                                  */
/*  char *   nazwa ścieżki do pliku w zintegrowanym systemie plików */
/*           zawierającego certyfikat                               */
/*  char *   etykieta dla certyfikatu                               */
/*                                                                  */
/*                                                                  */
/*                                                                  */
/* ---------------------------------------------------------------- */

#include <qydoadd1.h>
#include <stdlib.h>
#include <string.h>

int main (int argc, char *argv[])
{

    int         pathname_length, cert_label_length;
    Qus_EC_t    error_code;
    char      * pathname = argv[1];
    char      * certlabel = argv[2];

    /* szukanie dł. nazwy ścieżki */
    for(pathname_length = 0;
        ((*(pathname + pathname_length) != ' ') &&
         (*(pathname + pathname_length) != '\00'));
        pathname_length++);

    /* szukanie dł. etykiety certyfikatu */
    for(cert_label_length = 0;
        ((*(certlabel + cert_label_length) != ' ') &&
         (*(certlabel + cert_label_length) != '\00'));
        cert_label_length++);


    error_code.Bytes_Provided = 0;     /* wyjątki zwracane dla błędów       */

    QydoAddVerifier (pathname,         /* nazwa ść. do pliku z certyfikatem */
                     &pathname_length, /* długość nazwy ścieżki             */
                     "OBJN0100",       /* nazwa formatu                     */
                     certlabel,        /* etykieta certyfikatu              */
                     &cert_label_length, /* długość etykiety certyfikatu    */
                     &error_code);     /* kod błędu                         */

    return 0;
}
```

Po zakończeniu tych zadań możesz tworzyć pakiety swoich aplikacji i rozpowszechniać je wśród klientów. Podczas instalowania przez klientów podpisanych obiektów aplikacji zostaną one zweryfikowane w ramach procesu instalacji.

Klienci dysponują także możliwością weryfikowania podpisów na obiektach aplikacji za pomocą programu Digital Certificate Manager. Daje to klientom możliwość sprawdzenia, czy aplikacja pochodzi z zaufanego źródła i czy od czasu jej podpisania nie została zmieniona.

Uwaga: Program instalacyjny może utworzyć u klienta bazę certyfikatów *SIGNATUREVERIFICATION z hasłem domyślnym. Należy poinformować klienta, że powinien jak najszybciej za pomocą programu DCM zmienić hasło do bazy certyfikatów, aby ochronić ją przed nieuprawnionym dostępem.

Czynność 8: resetowanie domyślnego hasła bazy certyfikatów *SIGNATUREVERIFICATION przez klientów

Funkcja API Add Verifier mogła utworzyć bazę certyfikatów *SIGNATUREVERIFICATION w ramach instalowania produktu w systemie klienta. Jeśli funkcja ta tworzy bazę certyfikatów, to dla bazy powstaje równocześnie hasło domyślne. Należy więc doradzić klientowi skorzystanie z programu DCM do zmiany hasła, aby chronić bazę certyfikatów przed dostępem nieuprawnionych osób.

W tym celu do klientów kierowane są następujące polecenia:

1. Uruchom program DCM. Więcej informacji zawiera temat Uruchamianie programu DCM.
2. W ramce nawigacyjnej kliknij opcję **Wybór bazy certyfikatów**, a następnie wybierz ***SIGNATUREVERIFICATION** jako bazę certyfikatów, która ma zostać otworzona.
3. Gdy pojawi się strona Baza certyfikatów i hasło kliknij **Zerowanie hasła**, aby wyświetlić stronę Zerowanie hasła bazy certyfikatów.

   Uwaga: W przypadku pojawienia się pytań dotyczących określonego formularza w tym zadaniu należy kliknąć przycisk znaku zapytania (**?**) u góry strony, aby uzyskać dostęp do systemu pomocy.
4. Podaj nowe hasło dla tej bazy, wpisz je ponownie w celu potwierdzenia, wybierz strategię dla ważności hasła do bazy certyfikatów i kliknij **Kontynuuj**.

Scenariusz: podpisywanie obiektów za pomocą Centrum Zarządzania programu System i Navigator

W scenariuszu przedstawiono przedsiębiorstwo, które przesyła pakiety obiektów do wielu systemów i chce podpisywać te obiekty za pomocą odpowiednich funkcji systemu i5/OS. Sposób korzystania z funkcji Centrum Zarządzania programu System i Navigator w celu tworzenia i podpisywania pakietów, które następnie są przesyłane do innych systemów, został opisany w oparciu o potrzeby przedsiębiorstwa i jego cele związane z bezpieczeństwem.

Sytuacja

Przedsiębiorstwo (MojaFirma) tworzy aplikacje, które następnie przesyła do wielu systemów rozproszonych po całym przedsiębiorstwie. Jako administrator sieci, użytkownik jest odpowiedzialny za instalowanie i aktualizowanie tych aplikacji na wszystkich systemach przedsiębiorstwa. Użytkownik korzysta obecnie z funkcji Centrum Zarządzania programu System i Navigator, aby łatwiej tworzyć pakiety i przesyłać aplikacje oraz wykonywać inne zadania administracyjne, za które jest odpowiedzialny. Jednak śledzenie i rozwiązywanie problemów spowodowanych wprowadzonymi do tych aplikacji zmianami dokonanymi przez nieuprawnionych użytkowników zajmuje dużo czasu. Dlatego też chcesz lepiej chronić integralność tych obiektów i podpisywać je cyfrowo.

Przegląd możliwości podpisywania obiektów w systemie i5/OS pokazuje, że począwszy od wersji V5R2 Centrum Zarządzania umożliwia podpisywanie obiektów w trakcie tworzenia i przesyłania pakietów. Dzięki programowi Centrum Zarządzania możesz działać szybko i efektywnie, aby łatwo sprostać założonym celom ochrony przedsiębiorstwa. Zdecydowałeś się też na utworzenie lokalnego ośrodka CA, aby wystawiał certyfikaty do podpisywania obiektów. Wykorzystanie certyfikatów wydawanych przez lokalny ośrodek CA ogranicza wydatki związane z używaniem tej technologii ochrony, gdyż nie trzeba kupować certyfikatu od ogólnie znanego ośrodka CA.

Przykład ten jest praktycznym wprowadzeniem do konfigurowania i używania funkcji podpisywania obiektów aplikacji, które mają być przesłane do wielu systemów w przedsiębiorstwie.

Zalety scenariusza

Scenariusz ten ma następujące zalety:

- Korzystanie z Centrum Zarządzania w celu tworzenia pakietów i podpisywania obiektów pozwala skrócić czas potrzebny na przesyłanie podpisanych obiektów do systemów przedsiębiorstwa.
- Korzystanie z Centrum Zarządzania do podpisywania obiektów w pakietach zmniejsza ilość koniecznych działań, gdyż proces podpisywania jest częścią procesu tworzenia pakietów.
- Podpisywanie pakietu obiektów umożliwia łatwe określenie, czy obiekty zostały zmienione od czasu podpisania. Pozwala zmniejszyć nakłady na przyszłe rozwiązywanie i śledzenie problemów w aplikacjach.
- Wykorzystanie do podpisywania obiektów certyfikatu wystawionego przez prywatny ośrodek certyfikacji obniża koszty wprowadzenia podpisywania obiektów.

Cele

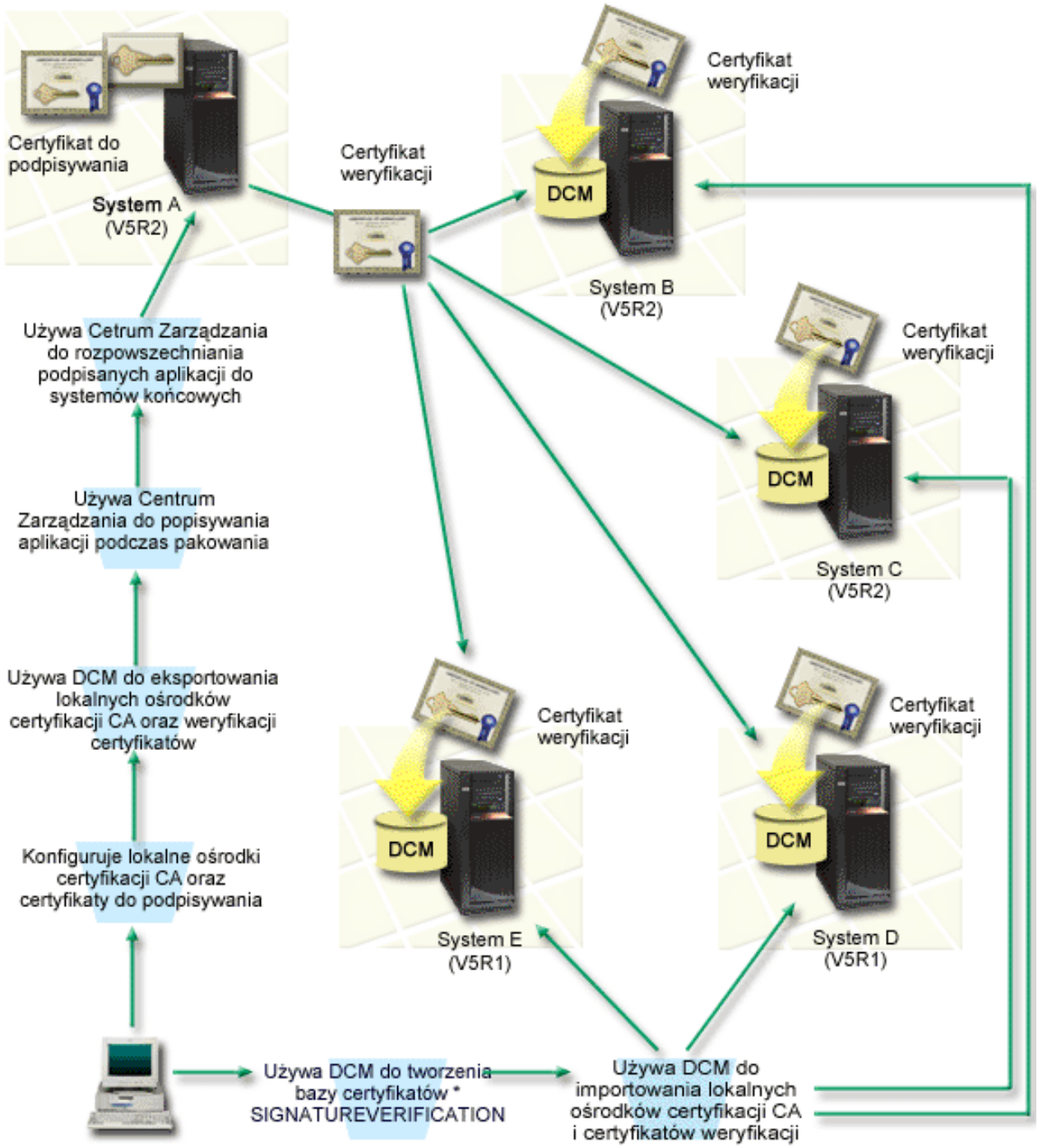
W ramach scenariusza zakładamy, że przedsiębiorstwo MojaFirma chce podpisywać cyfrowo aplikacje, które przesyła do wielu systemów w przedsiębiorstwie. Jako administrator sieci w przedsiębiorstwie MojaFirma, użytkownik korzysta z Centrum Zarządzania w celu wykonywania wielu zdań administracyjnych. Z tego względu użytkownik chce rozszerzyć obecne wykorzystanie Centrum Zarządzania o podpisywanie aplikacji przedsiębiorstwa przesyłanych do innych systemów.

Główne założenia scenariusza są następujące:

- Aplikacje przedsiębiorstwa muszą być podpisane certyfikatem wystawionym przez lokalny ośrodek CA, aby ograniczyć koszty podpisywania aplikacji.
- Administratorzy systemu i inni wyznaczeni użytkownicy muszą mieć możliwość prostego weryfikowania podpisów cyfrowych we wszystkich systemach, aby sprawdzić źródło i autentyczność obiektów podpisanych przez przedsiębiorstwo. Aby osiągnąć postawiony cel, każdy system musi mieć w bazie certyfikatów *SIGNATUREVERIFICATION zarówno kopię certyfikatu przedsiębiorstwa do weryfikowania podpisów, jak i certyfikat lokalnego ośrodka CA.
- Weryfikowanie podpisów aplikacji przedsiębiorstwa umożliwia administratorom i innym osobom wykrywanie zmian wprowadzonych w obiektach od czasu ich podpisania.
- Administratorzy muszą umieć posługiwać się Centrum Zarządzania w celu tworzenia pakietów, podpisywania, a następnie przesyłania aplikacji do systemów.

Informacje szczegółowe

Poniższy rysunek przedstawia proces podpisywania obiektu i weryfikowania podpisu według scenariusza:



Rysunek ilustruje następujące punkty scenariusza:

System centralny (System A)

- System A to serwer System i, na którym działa system operacyjny OS/400 Wersja 5, Wydanie 2 (V5R2).
- System A służy za system centralny, z którego uruchamiane są funkcje Centrum Zarządzania, w tym tworzenie pakietów i przesyłanie aplikacji przedsiębiorstwa.

- W systemie A zainstalowany jest produkt Cryptographic Access Provider 128-bit for System i (5722–AC3).
- W systemie A zainstalowano i skonfigurowano produkty Digital Certificate Manager (opcja 34) oraz IBM HTTP Server (5722–DG1).
- System A funkcjonuje jako lokalny ośrodek CA i w tym systemie znajdują się certyfikaty podpisujące obiekty.
- System A jest podstawowym systemem podpisującym obiekty dla aplikacji przedsiębiorstwa. Podpisywanie obiektów produktów przeznaczonych do dystrybucji wśród klientów przebiega w systemie A w następujący sposób:
  1. Użycie programu DCM do utworzenia lokalnego ośrodka CA, a następnie lokalnego ośrodka CA do utworzenia certyfikatu do podpisywania obiektów.
  2. Użycie programu DCM do wyeksportowania kopii certyfikatu lokalnego ośrodka CA i certyfikatu do weryfikowania podpisów plików, aby systemy końcowe (B, C, D i E) mogły weryfikować podpisane obiekty.
  3. Użycie programu Centrum Zarządzania do podpisania obiektów aplikacji i utworzenia z nich pakietów z plikami certyfikatów do weryfikowania.
  4. Użycie programu Centrum Zarządzania do rozpowszechnienia podpisanych aplikacji i plików certyfikatów do systemów końcowych.

Systemy końcowe (B, C, D i E)

- Systemy B i C to serwery System i, na których działa system operacyjny OS/400 Wersja 5, Wydanie 2 (V5R2).
- Systemy D i E to serwery System i, na których działa system OS/400 Wersja 5, Wydanie 1 (V5R1).
- W systemach B, C, D i E zainstalowano i skonfigurowano produkty Digital Certificate Manager (opcja 34) oraz IBM HTTP Server (5722–DG1).
- Systemy B, C, D i E otrzymują z systemu centralnego wraz z podpisaną aplikacją kopie obydwu certyfikatów, tzn. przeznaczonego do weryfikowania podpisów i pochodzącego od lokalnego ośrodka CA przedsiębiorstwa.
- program DCM służy do utworzenia bazy certyfikatów *SIGNATUREVERIFICATION i zaimportowania do niej certyfikatów lokalnego ośrodka CA i do weryfikacji podpisów.

Wymagania wstępne i założenia

Scenariusz zależy od spełnienia następujących założeń i wymagań wstępnych:

1. Wszystkie systemy spełniają wymagania w zakresie instalowania i używania programu Digital Certificate Manager (DCM).
2. W żadnym systemie nie był wcześniej konfigurowany ani używany program DCM.
3. System A spełnia wymagania niezbędne do zainstalowania i używania programu System i Navigator i Centrum Zarządzania.
4. We wszystkich systemach końcowych musi być uruchomiony serwer Centrum Zarządzania.
5. We wszystkich systemach zainstalowano najnowszą wersję programu licencjonowanego Cryptographic Access Provider 128-bit (5722-AC3).
6. Dla wartości systemowej Weryfikowanie podpisów obiektów podczas odtwarzania (VeriFY OBJect signatures during ReSTore - QVFYOBJRST) określono domyślną wartość 3 we wszystkich systemach objętych scenariuszem i nie została ona zmieniona. Ustawienie domyślne daje gwarancję, że system będzie mógł zweryfikować podpisy po odtworzeniu podpisanych obiektów.
7. Aby administrator sieci systemu A mógł podpisywać obiekty, musi mieć uprawnienia specjalne profilu użytkownika *ALLOBJ albo jego profil użytkownika musi być uprawniony do korzystania z aplikacji podpisującej obiekty.
8. Administrator sieci lub inna osoba, która tworzy bazę certyfikatów w programie DCM, musi mieć uprawnienia specjalne profilu użytkownika *SECADM i *ALLOBJ.
9. W celu weryfikowania podpisów obiektów administratorzy systemów lub inni użytkownicy wszystkich pozostałych systemów muszą mieć uprawnienia specjalne profilu użytkownika *AUDIT.

Czynności konfiguracyjne

Poniżej przedstawiono dwa zestawy zadań, które należy wykonać, aby wdrożyć ten scenariusz. Jeden zestaw zadań pozwala skonfigurować system A, aby używał on Centrum Zarządzania w celu podpisywania i przesyłania aplikacji. Drugi zestaw zadań umożliwia administratorom systemu i pozostałym użytkownikom weryfikowanie podpisów aplikacji na wszystkich pozostałych systemach. Poniżej zamieszczono szczegółowe informacje dotyczące wykonywania powyższych czynności.

Czynności związane z podpisywaniem obiektów

Informacje dotyczące podpisywania obiektów według opisu przedstawionego w ramach tego scenariusza zamieszczono poniżej w szczegółach do scenariusza. Zawierają one listę czynności umożliwiającą wykonanie każdego z następujących zadań w systemie A:

1. Spełnienie wszystkich wymagań wstępnych koniecznych do zainstalowania i skonfigurowania wszystkich potrzebnych produktów serwera System i.
2. Używanie programu DCM w celu utworzenia lokalnego ośrodka CA wystawiającego prywatny certyfikat podpisujący obiekt.
3. Używanie programu DCM w celu tworzenia definicji aplikacji.
4. Używanie programu DCM w celu przypisana certyfikatu do definicji aplikacji podpisującej obiekty.
5. Używanie programu DCM w celu eksportowania certyfikatów, których inne systemy muszą użyć do zweryfikowania podpisów obiektów. Konieczne jest wyeksportowanie do pliku zarówno kopii certyfikatu lokalnego ośrodka CA, jak i kopii certyfikatu do podpisywania obiektów, jako certyfikatu do weryfikowania podpisów.
6. Przesyłanie plików certyfikatów do każdego systemu końcowego, w którym mają być weryfikowane podpisy.
7. Używanie Centrum Zarządzania programu System i Navigator w celu podpisywania obiektów aplikacji.

Czynności związane z weryfikowaniem podpisów

Przed wysłaniem za pomocą Centrum Zarządzania podpisanych obiektów aplikacji do każdego z systemów końcowych należy skonfigurować w tych systemach weryfikowanie podpisów. Konfigurowanie weryfikowania podpisów musi zostać zakończone przed weryfikowaniem podpisów odtworzonych podpisanych obiektów na systemach końcowych.

W każdym systemie końcowym należy wykonać następujące czynności, aby zgodnie z założeniami scenariusza weryfikować podpisy obiektów:

1. Użyj programu DCM w celu utworzenia bazy certyfikatów *SIGNATUREVERIFICATION
2. Użyj programu DCM w celu zaimportowania certyfikatu lokalnego ośrodka CA i certyfikatu do weryfikowania podpisu

   Informacje pokrewne

   Digital Certificate Manager (DCM)

Szczegóły scenariusza: podpisywanie obiektów za pomocą Centrum Zarządzania programu System i Navigator

Ten temat zawiera informacje o konfigurowaniu Centrum Zarządzania i podpisywaniu za jego pomocą obiektów systemu operacyjnego i5/OS.

Czynność 1: spełnienie wszystkich wymagań wstępnych

Przed wykonaniem określonych zadań konfiguracyjnych związanych z implementacją tego scenariusza należy w całości spełnić wymagania wstępne w zakresie instalowania i konfigurowania wszystkich potrzebnych produktów serwera System i.

Czynność 2: tworzenie lokalnego ośrodka certyfikacji w celu wystawienia certyfikatu do podpisywania obiektu prywatnego

Podczas tworzenia lokalnego ośrodka CA za pomocą programu Digital Certificate Manager należy wypełnić szereg formularzy. Formularze te prowadzą przez proces tworzenia ośrodka CA i inne czynności niezbędne, aby rozpocząć korzystanie z certyfikatów cyfrowych w połączeniu z protokołem SSL, czyli podpisywanie obiektów i weryfikowanie podpisów. Wprawdzie w tym scenariuszu nie trzeba konfigurować certyfikatów w połączeniu z protokołem SSL, ale w celu skonfigurowania systemu do podpisywania obiektów należy wypełnić wszystkie formularze.

Aby użyć programu DCM w celu utworzenia i stosowania lokalnego ośrodka CA, wykonaj następujące czynności: po utworzeniu lokalnego ośrodka CA i certyfikatu podpisującego obiekt, przed podpisywaniem obiektów należy zdefiniować korzystającą z certyfikatu aplikację podpisującą obiekty.

1. Uruchom program DCM. Więcej informacji zawiera temat Uruchamianie programu DCM.
2. W ramce nawigacji programu DCM wybierz **Tworzenie ośrodka certyfikacji**, aby wyświetlić szereg formularzy.

   Uwaga: W przypadku pojawienia się pytań dotyczących określonego formularza w tym zadaniu należy kliknąć przycisk znaku zapytania (**?**) u góry strony, aby uzyskać dostęp do systemu pomocy.
3. Wypełnij wszystkie formularze zadań. Po zakończeniu tego zadania, wykonaj poniższe czynności:
   a. Wprowadź informacje identyfikujące lokalny ośrodek CA.
   b. Zainstaluj certyfikat lokalnego ośrodka CA w swojej przeglądarce, aby oprogramowanie mogło go rozpoznać i sprawdzać poprawność certyfikatów wystawionych przez ten ośrodek.
   c. Zdefiniuj strategię lokalnego ośrodka CA.
   d. Użyj nowego lokalnego ośrodka CA do wystawienia certyfikatu klienta lub serwera, z którego aplikacja będzie mogła skorzystać do połączeń z użyciem protokołu SSL.

      Uwaga: Wprawdzie opisywany scenariusz nie korzysta z tego certyfikatu, jego utworzenie jest jednak niezbędne przed użyciem lokalnego ośrodka CA do wystawienia potrzebnego certyfikatu podpisującego obiekt. Jeśli zadanie zostanie anulowane bez utworzenia certyfikatu, to należy utworzyć certyfikat podpisujący obiekt i bazę certyfikatów *OBJECTSIGNING, w której będzie on przechowywany.
   e. Wybierz aplikacje, które będą mogły używać certyfikatu serwera lub klienta w połączeniach SSL.

      Uwaga: Na potrzeby tego scenariusza nie wybieraj żadnej aplikacji, tylko kliknij **Kontynuuj**, aby wyświetlić następny formularz.
   f. Użyj nowego lokalnego ośrodka CA do wystawienia certyfikatu podpisującego obiekt, z którego aplikacja będzie mogła skorzystać do cyfrowego podpisywania obiektów. W tym podzadaniu tworzona jest baza certyfikatów *OBJECTSIGNING. Jest to baza umożliwiająca zarządzanie certyfikatami do podpisywania obiektów.
   g. Wybierz aplikacje, które powinny ufać lokalnemu ośrodkowi CA.

      Uwaga: W ramach tego scenariusza nie wybieraj żadnych aplikacji i kliknij przycisk **Kontynuuj**, aby zakończyć zadanie.

Czynność 3: tworzenie definicji aplikacji podpisującej obiekty

Po utworzeniu certyfikatu podpisującego obiekt należy za pomocą programu Digital Certificate Manager utworzyć definicję aplikacji podpisującej obiekty, która będzie używana do podpisywania obiektów. Definicja aplikacji nie musi odnosić się do istniejącej aplikacji. Tworzona definicja aplikacji powinna opisywać typ lub grupę obiektów, które zamierzasz podpisywać. Definicja jest niezbędna, żeby można było powiązać identyfikator aplikacji z certyfikatem i umożliwić proces podpisywania.

Aby utworzyć definicję aplikacji podpisującej obiekty za pomocą programu DCM, wykonaj następujące czynności:

1. W ramce nawigacyjnej kliknij opcję **Wybór bazy certyfikatów** i wybierz ***OBJECTSIGNING** jako bazę certyfikatów, która ma zostać otworzona.
2. Po wyświetleniu strony Baza certyfikatów i hasło wpisz hasło określone dla bazy certyfikatów przy jej tworzeniu i kliknij przycisk **Kontynuuj**.
3. W ramce nawigacji wybierz **Zarządzanie aplikacjami**, aby wyświetlić listę zadań.
4. Z listy zadań wybierz **Dodaj aplikację**, aby wyświetlić formularz definiowania aplikacji.
5. Wypełnij formularz i kliknij **Dodaj**.

Należy teraz przypisać certyfikat podpisujący obiekt do utworzonej aplikacji.

Czynność 4: przypisanie certyfikatu do definicji aplikacji podpisującej obiekty

Aby przypisać certyfikat do aplikacji podpisującej obiekty, wykonaj następujące czynności:

1. W ramce nawigacji programu DCM wybierz **Zarządzanie certyfikatami**, aby wyświetlić listę zadań.
2. Z listy tej wybierz **Przypisanie certyfikatu**, aby wyświetlić listę certyfikatów bieżącej bazy certyfikatów.
3. Wybierz certyfikat z listy i kliknij **Przypisz do aplikacji**, aby wyświetlić listę definicji aplikacji bieżącej bazy certyfikatów.
4. Wybierz jedną lub więcej aplikacji z listy i kliknij **Kontynuuj**. Pojawi się strona komunikatów, przedstawiająca albo potwierdzenie przypisania certyfikatu, albo informacje o błędzie, jeśli wystąpił jakiś problem.

Po zakończeniu tych czynności możesz korzystać z podpisywania obiektów za pomocą programu Centrum Zarządzania podczas tworzenia pakietów i ich rozpowszechniania. Aby jednak mieć pewność, że użytkownik i inne osobą mogą weryfikować podpisy, należy wyeksportować niezbędne certyfikaty do pliku i przesłać je do wszystkich systemów końcowych. Przed wysłaniem za pomocą Centrum Zarządzania podpisanych obiektów aplikacji do każdego z systemów końcowych należy skonfigurować w tych systemach weryfikowanie podpisów. Konfigurowanie weryfikowania podpisów musi zostać zakończone przed weryfikowaniem podpisów odtworzonych podpisanych obiektów na systemach końcowych.

Czynność 5: eksportowanie certyfikatów w celu umożliwienia weryfikowania podpisów w innych systemach

Podpisywanie obiektów w celu ochrony integralności ich zawartości ma sens tylko wtedy, gdy istnieją metody weryfikowania autentyczności podpisu. Aby zweryfikować podpisy obiektów na tym samym systemie, który podpisuje obiekty, należy użyć programu DCM do utworzenia bazy certyfikatów *SIGNATUREVERIFICATION. Musi ona zawierać zarówno kopię certyfikatu podpisującego obiekt, jak i kopię certyfikatu ośrodka CA, który wystawił certyfikat podpisujący.

Aby inni użytkownicy mogli weryfikować podpis, należy im dostarczyć kopię certyfikatu podpisującego obiekt. Jeśli do wystawienia certyfikatu korzysta się z lokalnego ośrodka CA, trzeba im zapewnić także kopię certyfikatu lokalnego ośrodka CA.

Aby za pomocą programu DCM weryfikować podpisy w tym samym systemie, który podpisuje obiekty (w tym scenariuszu jest to system A), wykonaj następujące czynności:

1. W oknie nawigacji wybierz **Tworzenie nowej bazy certyfikatów**, a następnie, jako bazę certyfikatów do utworzenia wybierz ***SIGNATUREVERIFICATION**.
2. Wybierz **Tak**, aby skopiować do nowej bazy certyfikatów istniejące certyfikaty podpisujące obiekty jako certyfikaty weryfikujące podpisy.
3. Podaj hasło dla nowej bazy certyfikatów i kliknij **Kontynuuj**, aby ją utworzyć. Od tego momentu za pomocą programu DCM możesz weryfikować podpisy obiektów na tym samym systemie, którego używasz do podpisywania obiektów.

Aby użyć programu DCM do eksportowania kopii certyfikatu lokalnego CA i kopii certyfikatu podpisującego obiekty jako certyfikatu weryfikującego podpisy, w celu weryfikacji podpisów obiektów na innych systemach, wykonaj następujące czynności:

1. W ramce nawigacji wybierz **Zarządzanie certyfikatami**, a następnie wybierz zadanie **Eksport certyfikatu**.
2. Wybierz opcję **Ośrodek certyfikacji (CA)** i kliknij przycisk **Kontynuuj**, aby wyświetlić listę certyfikatów CA, które można wyeksportować.
3. Wybierz z listy uprzednio utworzony certyfikat lokalnego CA i kliknij **Eksportuj**.
4. Określ **Plik** jako miejsce docelowe eksportowania i kliknij przycisk **Kontynuuj**.
5. Podaj pełną ścieżkę i nazwę dla pliku eksportowanego certyfikatu lokalnego CA i kliknij **Kontynuuj**, aby wyeksportować certyfikat.
6. Kliknij **OK**, aby opuścić stronę potwierdzenia eksportu. Możesz już eksportować kopię certyfikatu podpisującego obiekty.
7. Ponownie wybierz zadanie **Eksportuj certyfikat**.
8. Wybierz **Podpisywanie obiektów**, aby wyświetlić listę certyfikatów podpisujących obiekty, które można eksportować.
9. Wybierz z listy odpowiedni certyfikat podpisujący obiekty i kliknij opcję **Eksportuj**.
10. Wybierz **Plik, jako certyfikat do weryfikowania podpisów** jako miejsce docelowe i kliknij **Kontynuuj**.
11. Podaj pełną ścieżkę i nazwę dla pliku eksportowanego certyfikatu do weryfikowania podpisów i kliknij **Kontynuuj**, aby wyeksportować certyfikat.

Od tego momentu można przesyłać te pliki do systemów końcowych, na których mają być weryfikowane podpisy utworzone za pomocą certyfikatu.

Czynność 6: przesyłanie plików certyfikatów do systemów końcowych

Przed skonfigurowaniem systemów w celu weryfikowania podpisanych obiektów należy przesłać pliki certyfikatów utworzone w systemie A do systemów końcowych w ramach tego scenariusza. Do przesłania plików certyfikatów można użyć kilku metod, na przykład skorzystać z protokołu FTP lub z rozpowszechniania pakietów w programie Centrum Zarządzania.

Czynność 7: podpisywanie obiektów przy użyciu Centrum Zarządzania

Proces podpisywania obiektów jest dla programu Centrum Zarządzania częścią procesu dystrybucji pakietów oprogramowania. Przed wysłaniem za pomocą Centrum Zarządzania podpisanych obiektów aplikacji do każdego z systemów końcowych należy skonfigurować w tych systemach wszystkie zadania związane z konfigurowaniem weryfikowania podpisów. Konfigurowanie weryfikowania podpisów musi zostać zakończone przed weryfikowaniem podpisów odtworzonych podpisanych obiektów na systemach końcowych.

Aby zgodnie z założeniami scenariusza podpisać aplikację przesyłaną do systemów końcowych, należy wykonać następujące czynności:

1. Użyj programu Centrum Zarządzania do utworzenia pakietu i dystrybucji oprogramowania.
2. Po przejściu do panelu **Identyfikacja** kreatora **Definicja produktu** kliknij opcję **Zaawansowane**, aby wyświetlić panel **Zaawansowana identyfikacja**.
3. W polu **Cyfrowo podpisuje** wprowadź identyfikator wcześniej utworzonej aplikacji podpisującej i kliknij **OK**.
4. Zakończ kreatora i kontynuuj proces tworzenia pakietów i dystrybucji oprogramowania w programie Centrum Zarządzania.

Czynność 8: zadania z zakresu weryfikowania podpisu: tworzenie bazy certyfikatów *SIGNATUREVERIFICATION w systemach końcowych

Aby zgodnie ze scenariuszem weryfikować podpisy obiektów na systemach końcowych, każdy system musi mieć kopię odpowiedniego certyfikatu do weryfikowania podpisów w swojej bazie certyfikatów

*SIGNATUREVERIFICATION. Jeśli obiekt był podpisany certyfikatem prywatnym, w bazie certyfikatów musi również znaleźć się kopia tego certyfikatu lokalnego ośrodka CA.

Aby utworzyć bazę certyfikatów *SIGNATUREVERIFICATION, wykonaj następujące czynności:

1. Uruchom program DCM. Więcej informacji zawiera temat Uruchamianie programu DCM.
2. W ramce nawigacyjnej programu Digital Certificate Manager wybierz opcję **Tworzenie nowej bazy certyfikatów**, a następnie wybierz ***SIGNATUREVERIFICATION** jako nową bazę certyfikatów.

   Uwaga: W przypadku pojawienia się pytań dotyczących określonego formularza w tym zadaniu należy kliknąć przycisk znaku zapytania (**?**) u góry strony, aby uzyskać dostęp do systemu pomocy.
3. Podaj hasło dla nowej bazy certyfikatów i kliknij **Kontynuuj**, aby ją utworzyć. Możesz teraz importować certyfikaty do bazy i korzystać z nich do weryfikowania podpisów obiektów.

Czynność 9: zadania z zakresu weryfikowania podpisu: importowanie certyfikatów

Aby weryfikować podpis na obiekcie, baza *SIGNATUREVERIFICATION musi zawierać kopię certyfikatu do weryfikowania podpisów. Jeśli certyfikat podpisujący jest prywatny, to w bazie certyfikatów musi znaleźć się również kopia certyfikatu lokalnego ośrodka CA, który go wystawił. W opisywanym scenariuszu obydwa certyfikaty zostały wyeksportowane do pliku, który został przesłany do każdego systemu końcowego.

Wykonaj następujące czynności, aby zaimportować te certyfikaty do bazy *SIGNATUREVERIFICATION: Podczas odtwarzania podpisanych obiektów system będzie teraz mógł weryfikować podpisy obiektów, które zostały utworzone za pomocą odpowiedniego certyfikatu podpisującego.

1. W oknie nawigacji programu DCM kliknij **Wybór ośrodka certyfikacji** i wybierz ***SIGNATUREVERIFICATION** jako bazę certyfikatów do otwarcia.
2. Po wyświetleniu strony Baza certyfikatów i hasło wpisz hasło określone dla bazy certyfikatów przy jej tworzeniu i kliknij przycisk **Kontynuuj**.
3. Po odświeżeniu widoku ramki nawigacji wybierz opcję **Zarządzanie certyfikatami**, aby wyświetlić listę zadań.
4. Z listy zadań wybierz **Import certyfikatu**.
5. Jako typ certyfikatu wybierz **Ośrodek certyfikacji** i kliknij **Kontynuuj**.

   Uwaga: Przed zaimportowaniem prywatnego certyfikatu do weryfikowania podpisów należy zaimportować certyfikat lokalnego ośrodka CA, inaczej proces importu certyfikatu do weryfikowania podpisów nie powiedzie się.
6. Podaj pełną ścieżkę i nazwę dla pliku certyfikatu ośrodka CA i kliknij **Kontynuuj**. Pojawi się komunikat potwierdzający pomyślne zakończenie importu lub informacja o błędzie, jeśli proces nie powiódł się.
7. Ponownie wybierz zadanie **Importuj certyfikat**.
8. Jako typ certyfikatu wybierz **Sprawdzania podpisu** i kliknij **Kontynuuj**.
9. Podaj pełną ścieżkę i nazwę dla pliku certyfikatu do weryfikowania podpisów i kliknij **Kontynuuj**. Pojawi się komunikat potwierdzający pomyślne zakończenie importu lub informacja o błędzie, jeśli proces nie powiódł się.

Wymagania wstępne dotyczące podpisywania obiektów i weryfikowania podpisów

Ten temat zawiera informacje o wymaganiach wstępnych konfiguracji oraz innych założeniach dotyczących podpisywania obiektów i weryfikowania podpisów w systemie, w którym działa system operacyjny i5/OS.

Możliwości podpisywania obiektów i weryfikowania podpisów systemu i5/OS dostarczają silnych dodatkowych narzędzi służących do nadzorowania obiektów w systemie. Aby skorzystać z tych możliwości, należy spełnić następujące wymagania wstępne.

Wymagania wstępne dotyczące podpisywania obiektów

W zależności od wymagań firmy i ochrony możesz wybrać spośród kilku metod podpisywania obiektów:

- użyj programu Digital Certificate Manager (DCM),
- napisz program korzystający z funkcji API Sign Object ,
- użyj funkcji Centrum Zarządzania programu iSeries Navigator, aby podpisywać obiekty w trakcie tworzenia pakietów przeznaczonych do przesłania do systemów końcowych.

Wybór metody podpisywania obiektów zależy od wymagań firmy i oczekiwanej ochrony. Niezależnie jednak od planowanej metody podpisywania obiektów, należy spełnić pewne wymagania wstępne:

- Należy zrealizować wymagania wstępne niezbędne do zainstalowania i korzystania z programu Digital Certificate Manager (DCM).
  - Następnie trzeba przy użyciu programu DCM utworzyć bazę certyfikatów *OBJECTSIGNING. Tworzenie bazy certyfikatów może być częścią procesu tworzenia lokalnego ośrodka CA lub częścią procesu zarządzania certyfikatami podpisującymi obiekty pochodzącymi od publicznego, internetowego ośrodka CA.
  - Baza certyfikatów *OBJECTSIGNING musi zawierać przynajmniej jeden certyfikat, który może być utworzony przez lokalny ośrodek CA lub otrzymany od publicznego internetowego ośrodka CA.
  - Należy użyć programu DCM do utworzenia przynajmniej jednej definicji aplikacji podpisującej obiekt, używanej do podpisywania obiektów.
  - Należy użyć programu DCM do przypisania określonego certyfikatu do definicji aplikacji podpisującej obiekt.
- Profil użytkownika wykorzystywany do podpisywania obiektów musi mieć uprawnienia specjalne *ALLOBJ. Profil użytkownika wykorzystywany do tworzenia bazy certyfikatów *SIGNATUREVERIFICATION musi mieć uprawnienia specjalne *SECADM i *ALLOBJ.

Wymagania wstępne dotyczące weryfikowania podpisów

Istnieje kilka metod, których możesz użyć do weryfikowania podpisów na obiektach:

- użyj programu Digital Certificate Manager (DCM),
- napisz program korzystający z funkcji API Verify Object (QYDOVFYO),
- wybierz spośród komend na przykład komendę Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG).

Wybór metody weryfikacji podpisów zależy od wymagań firmy i ochrony. Niezależnie jednak od planowanej metody weryfikacji należy spełnić następujące wymagania wstępne:

- Należy zrealizować wymagania wstępne niezbędne do zainstalowania i korzystania z programu Digital Certificate Manager (DCM).
- Należy utworzyć bazę certyfikatów *SIGNATUREVERIFICATION. W zależności od potrzeb, można skorzystać z dwóch metod. Można użyć programu Digital Certificate Manager (DCM) do zarządzania certyfikatami do weryfikacji podpisów. Jeśli do podpisywania obiektów korzysta się z certyfikatów publicznych, można utworzyć bazę certyfikatów pisząc program korzystający z funkcji API Add Verifier (QYDOADDV).

  Uwaga: Funkcja ta tworzy bazę certyfikatów z domyślnym hasłem. Następnie należy użyć programu DCM do zmiany tego hasła, aby zapobiec nieuprawnionemu dostępowi do bazy certyfikatów.
- Baza *SIGNATUREVERIFICATION musi zawierać kopię certyfikatu, którym podpisano obiekty. Certyfikat można dodać do bazy na dwa sposoby. Można za pomocą programu DCM systemu podpisującego wyeksportować certyfikat do pliku i następnie, za pomocą tego samego programu na docelowym systemie weryfikującym, zaimportować certyfikat do bazy certyfikatów *SIGNATUREVERIFICATION. Można również, jeśli do podpisywania obiektów używa się certyfikatów publicznych, dodać certyfikat do bazy certyfikatów docelowego systemu weryfikującego za pomocą programu korzystającego z funkcji API Add Verifier.
- Baza *SIGNATUREVERIFICATION musi zawierać kopię certyfikatu ośrodka certyfikacji, z którego pochodzi certyfikat użyty do podpisania obiektów. Jeśli do podpisywania obiektów korzysta się z certyfikatu publicznego, to baza certyfikatów docelowego systemu weryfikującego może już zawierać kopię żądanego certyfikatu ośrodka CA.

Jeśli do podpisywania obiektów korzysta się z certyfikatu wystawionego przez lokalny ośrodek CA, to należy za pomocą programu DCM dodać na docelowym systemie weryfikującym kopię certyfikatu lokalnego ośrodka CA do bazy certyfikatów.

Uwaga: Ze względów bezpieczeństwa funkcja API Add Verifier nie pozwoli na dodanie do bazy certyfikatów *SIGNATUREVERIFICATION certyfikatu ośrodka certyfikacji. W momencie dodawania certyfikatu ośrodka certyfikacji do bazy certyfikatów system zakłada, że ośrodek CA jest zaufanym źródłem certyfikatów. W konsekwencji system traktuje certyfikat wystawiony przez ośrodek CA jako pochodzący z zaufanego źródła. Dlatego nie można korzystać z funkcji API w celu utworzenia programu instalacyjnego obsługi wyjścia w celu dodania certyfikatu ośrodka certyfikacji do bazy certyfikatów. Aby dodać certyfikat ośrodka CA do bazy certyfikatów, należy użyć programu Digital Certificate Manager; takie rozwiązanie daje pewność, że sprawowana jest ręczna, dokładna kontrola nad ośrodkami CA, którym system ufa. Dzięki temu system nie może importować certyfikatów ze źródeł, których administrator świadomie nie określił jako zaufane.

Jeśli do podpisywania obiektów użytkownik używa certyfikatu wystawionego przez lokalny ośrodek CA, należy za pomocą programu DCM w systemie lokalnego ośrodka CA wyeksportować kopię tego certyfikatu do pliku. Następnie można użyć programu DCM na docelowym systemie weryfikującym, aby zaimportować certyfikat lokalnego ośrodka CA do bazy certyfikatów *SIGNATUREVERIFICATION. Aby zapobiec możliwym błędom, należy zaimportować ten certyfikat przed użyciem funkcji API Add Verifier do dodania certyfikatu do weryfikowania podpisów. Dlatego, jeśli korzysta się z certyfikatu wystawionego przez lokalny ośrodek CA, można uznać za łatwiejsze użycie programu DCM do importu do bazy certyfikatów zarówno certyfikatu ośrodka CA, jak i certyfikatu weryfikującego.

Jeśli nie chcesz, aby ktokolwiek mógł dodać certyfikat weryfikacji do bazy certyfikatów *SIGNATUREVERIFICATION, korzystając z tej funkcji API, zablokuj tę funkcję w systemie. Aby zrealizować to zadanie, skorzystaj z systemowych narzędzi serwisowych (SST) umożliwiających ustawienie odrzucania zmian wprowadzonych w wartościach systemowych związanych z ochroną.

- Profil użytkownika systemu wykorzystywany do weryfikowania podpisów musi mieć uprawnienia specjalne *AUDIT. Profil użytkownika systemu wykorzystywany do tworzenia bazy certyfikatów lub zmiany hasła dla tej bazy musi mieć uprawnienia specjalne *SECADM i *ALLOBJ.

Zarządzanie podpisanymi obiektami

Ten temat zawiera informacje o komendach i wartościach systemowych używanych w systemie operacyjnym i5/OS do pracy z podpisanymi obiektami. Ponadto opisano w nim wpływ podpisanych obiektów na procesy składowania i odtwarzania.

Począwszy od wydania V5R1, firma IBM podpisuje programy licencjonowane systemu i5/OS oraz poprawki PTF, znakując w ten sposób oficjalnie system operacyjny jako pochodzący od firmy IBM, co umożliwia wykrycie nieautoryzowanych zmian w obiektach systemowych. Partnerzy handlowi i inni sprzedawcy mogą także podpisywać dostarczane przez siebie aplikacje. Dlatego nawet jeśli nie podpisujesz samodzielnie obiektów, musisz wiedzieć, jak pracować z podpisanymi obiektami i jak one wpływają na rutynowe zadania administracyjne systemu.

Przede wszystkim podpisane obiekty wpływają na zadania składowania i odtwarzania, a dokładnie na to, jak obiekty są składowane i odtwarzane w systemie.

Wartości systemowe i komendy wpływające na podpisane obiekty

Ten temat zawiera informacje o wartościach systemowych i komendach systemu operacyjnego i5/OS, które służą do zarządzania podpisanymi obiektami lub wpływają na nie po uruchomieniu.

Aby efektywnie zarządzać podpisanymi obiektami, należy zrozumieć jak wartości systemowe i komendy wpływają na podpisane obiekty. Wartość systemowa **Weryfikowanie podpisów obiektów podczas odtwarzania** (QVFYOBJRST) określa, w jaki sposób różne komendy odtwarzania wpływają na podpisane obiekty i jak system obsłuży podpisane obiekty w trakcie operacji odtwarzania. W systemie nie istnieją komendy języka CL przeznaczone wyłącznie do pracy

z podpisanymi obiektami. Jednak istnieją pewne wspólne komendy, których można użyć do zarządzania podpisanymi obiektami (lub do zarządzania obiektami infrastruktury umożliwiającymi podpisywanie obiektów). Inne komendy mogą niekorzystnie wpłynąć na podpisane obiekty w systemie, poprzez usunięcie podpisu z obiektu i tym samym zlikwidowanie ochrony, jakiej ten podpis dostarczał.

Wartości systemowe wpływające na podpisane obiekty

Wartość systemowa **Weryfikowanie podpisów obiektów podczas odtwarzania** (Verify object signatures during restore - QVFYOBJRST), należąca do kategorii wartości systemowych dotyczących odtwarzania w systemie i5/OS określa, w jaki sposób komendy w systemie wpływają na podpisane obiekty. Ta wartość systemowa, dostępna za pomocą programu iSeries Navigator, steruje obsługą weryfikowania podpisów przez system podczas operacji odtwarzania. Ustawienia tej wartości systemowej, wraz z ustawieniami dwóch innych wartości systemowych wpływają na operacje odtwarzania w systemie. W zależności od wybranego ustawienia tej wartości, może ona zezwalać lub zabraniać odtwarzania obiektów na podstawie statusu ich podpisu (na przykład tego, czy obiekt nie jest podpisany, ma niepoprawny podpis, został podpisany przez zaufane źródło itp.) Domyślne ustawienie tej wartości systemowej umożliwia odtwarzanie niepodpisanych obiektów, ale zapewnia jednocześnie, że obiekty podpisane mogą być odtworzone tylko wtedy, gdy ich podpis jest prawidłowy. System określa obiekt jako podpisany tylko wtedy, gdy ma on podpis ośrodka certyfikacji, któremu system ufa; system ignoruje inne ″niewiarygodne″ podpisy obiektów i traktuje te obiekty jako niepodpisane.

Istnieje kilka ustawień, których można użyć dla wartości systemowej QVFYOBJRST, począwszy od ignorowania wszystkich podpisów, aż do wymagania prawidłowych podpisów dla wszystkich obiektów odtwarzanych w systemie. Wartość ta dotyczy jedynie odtwarzanych obiektów wykonywalnych, takich jak programy (*PGM), komendy (*CMD), programy serwisowe (*SRVPGM), pakiety SQL (*SQLPKG) czy moduły (*MODULE). Dotyczy także obiektów zbiorów strumieniowych (*STMF) powiązanych z programami w języku Java utworzonych za pomocą komendy Utwórz program Java (Create Java Program - CRTJVAPGM). Nie dotyczy natomiast plików systemu IFS ani zbiorów składowania (*SAV).

Komendy języka CL wpływające na podpisane obiekty

Istnieje kilka komend języka CL umożliwiających pracę z podpisanymi obiektami lub mających wpływ na podpisane obiekty w systemie. Można skorzystać z kilku komend do podglądania informacji o podpisie obiektu, sprawdzania podpisu na obiekcie i składowania oraz odtwarzania obiektów ochrony niezbędnych do weryfikowania podpisów. Ponadto istnieje grupa komend, których uruchomienie może usunąć podpis z obiektu, usuwając w ten sposób ochronę, jakiej ten podpis dostarczał.

Komendy służące do wyświetlania informacji o podpisie obiektu

- Komenda Wyświetl opis obiektu (Display Object Description - DSPOBJD) wyświetla nazwy i atrybuty określonych obiektów w określonej bibliotece lub w bibliotekach z listy bibliotek wątku. Za pomocą tej komendy można określić, czy obiekt został podpisany i przejrzeć informacje o podpisie.
- Komendy zintegrowanego systemu plików, Wyświetlenie dowiązań obiektu (Display Object Links - DSPLNK) i Praca z dowiązaniami obiektów (Work with Object Links - WRKLNK). Komend tych można użyć do wyświetlenia informacji o podpisie na obiekcie znajdującym się w zintegrowanym systemie plików.

Komendy służące do weryfikowania podpisów obiektów

- Komenda Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG) Umożliwia określenie, czy została naruszona integralność obiektów w systemie. Komendy tej można użyć do weryfikacji podpisów w podobny sposób, jak używa się programu antywirusowego do określenia, czy wirus uszkodził jakieś pliki lub inne obiekty w systemie. Więcej informacji o korzystaniu z tej komendy z podpisanymi obiektami zawiera artykuł Komendy sprawdzające kod i integralność podpisu.
- Komenda Sprawdzenie opcji produktu (Check Product Option - CHKPRDOPT) Komenda ta przedstawia różnice pomiędzy prawidłową a bieżącą strukturą oprogramowania. Na przykład pokaże błąd, jeśli z zainstalowanego produktu zostanie usunięty jakiś obiekt. Parametru CHKSIG można użyć do określenia, w jaki sposób komenda obsługuje i zgłasza ewentualne problemy z podpisem danego produktu. Więcej informacji o korzystaniu z tej komendy z podpisanymi obiektami zawiera artykuł Komendy sprawdzające kod i integralność podpisu.

- Komenda Składowanie programu licencjonowanego (Save Licensed Program - SAVLICPGM) Komenda składuje kopie obiektów tworzących program licencjonowany. Składuje go w takiej postaci, z której może zostać odtworzony komendą Odtworzenie programu licencjonowanego (Restore Licensed Program - RSTLICPGM). Parametru CHKSIG można użyć do określenia, w jaki sposób komenda obsługuje i zgłasza ewentualne problemy z podpisem danego produktu. Więcej informacji o korzystaniu z tej komendy z podpisanymi obiektami zawiera artykuł Komendy sprawdzające kod i integralność podpisu.
- Komenda Odtworzenie (Restore - RST) Komenda odtwarza kopię jednego lub większej liczby obiektów, z których można korzystać w zintegrowanym systemie plików. Umożliwia także odtworzenie baz certyfikatów i ich zawartości. Nie można jednak jej użyć do odtworzenia bazy certyfikatów *SIGNATUREVERIFICATION. Sposób obsługi przez tę komendę podpisanych obiektów jest określony przez ustawienie wartości systemowej Verify object signatures during restore (QVFYOBJRST).
- Komenda Odtworzenie biblioteki (Restore Library - RSTLIB) Komenda odtwarza bibliotekę lub grupę bibliotek składowanych komendą Składowanie biblioteki (Save Library - SAVLIB). Komenda odtwarza całą bibliotekę, włącznie z opisem biblioteki, opisem obiektu i zawartością obiektów w bibliotece. Sposób obsługi przez tę komendę podpisanych obiektów jest określony przez ustawienie wartości systemowej Verify object signatures during restore (QVFYOBJRST).
- Komenda Odtworzenie programu licencjonowanego (Restore Licensed Program - RSTLICPGM) Komenda ładuje i odtwarza programy licencjonowane dla instalacji początkowej lub instalacji nowej wersji. Sposób obsługi przez tę komendę podpisanych obiektów jest określony przez ustawienie wartości systemowej Verify object signatures during restore (QVFYOBJRST).
- Komenda Odtworzenie obiektu (Restore object - RSTOBJ) Komenda odtwarza jeden lub więcej obiektów pojedynczej biblioteki zapisanych na dyskietce, taśmie, nośniku optycznym lub w zbiorze składowania za pomocą pojedynczej komendy. Sposób obsługi przez tę komendę podpisanych obiektów jest określony przez ustawienie wartości systemowej Verify object signatures during restore (QVFYOBJRST).

Komendy do składowania i odtwarzania baz certyfikatów

- Komenda Składowanie (Save - SAV) Komenda umożliwia składowanie kopii jednego lub większej liczby obiektów, z których można korzystać w systemie plików IFS, włącznie z bazami certyfikatów. Nie można jednak jej użyć do składowania bazy certyfikatów *SIGNATUREVERIFICATION.
- Komenda Składowanie danych ochrony (Save Security Data - SAVSECDTA) Komenda umożliwia składowanie wszystkich informacji o ochronie bez potrzeby przenoszenia systemu w stan zastrzeżony. Komenda umożliwia składowanie bazy certyfikatów *SIGNATUREVERIFICATION wraz z zawartymi w niej certyfikatami. Nie składuje jednak innych baz certyfikatów.
- Komenda Składowanie systemu (Save System - SAVSYS) Komenda umożliwia składowanie kopii licencjonowanego kodu wewnętrznego i biblioteki QSYS w formacie zgodnym z zainstalowanym systemem. Nie składuje obiektów z żadnej innej biblioteki. Umożliwia ponadto składowanie obiektów ochrony i konfiguracyjnych, które można również składować komendami SAVSECDTA i SAVCFG. Komenda umożliwia składowanie bazy certyfikatów *SIGNATUREVERIFICATION wraz z zawartymi w niej certyfikatami.
- Komenda Odtworzenie (Restore - RST) Umożliwia odtworzenie baz certyfikatów i ich zawartości. Nie można jednak jej użyć do odtworzenia bazy certyfikatów *SIGNATUREVERIFICATION.
- Komenda Odtworzenie profili użytkowników (Restore User Profiles - RSTUSRPRF) Komenda umożliwia odtworzenie podstawowych części profilu użytkownika lub ustawienie profili użytkowników składowanych za pomocą komend Składowanie systemu (Save System - SAVSYS) lub Składowanie danych ochrony (Save Security Data - SAVSECDTA). Za pomocą tej komendy można odtworzyć bazę certyfikatów *SIGNATUREVERIFICATION i ukryte hasła dla tej bazy oraz innych baz certyfikatów. Można odtworzyć bazę certyfikatów *SIGNATUREVERIFICATION bez odtwarzania informacji o profilach użytkowników podając *DCM jako wartość parametru SECDTA i *NONE jako wartość parametru USRPRF. Aby za pomocą tej komendy odtworzyć informacje o profilach użytkowników i bazy certyfikatów wraz z hasłami, należy podać *ALL jako parametr USRPRF.

Komendy służące do usuwania podpisów z obiektów

Korzystając z tych komend można usunąć podpis z obiektu. Usunięcie podpisu może spowodować problemy z obiektem, z którego podpis został usunięty. W ostateczności nie będzie można zweryfikować źródła obiektu jako

zaufanego lub zweryfikować podpisu w celu wykrycia zmian w obiekcie. Komend tych należy używać wyłącznie w odniesieniu do obiektów podpisanych utworzonych samodzielnie (w odróżnieniu do obiektów podpisanych otrzymanych z innych źródeł, na przykład od firmy IBM lub od dostawców). Jeśli istnieją podejrzenia, że komenda usunęła podpis obiektu, można użyć komendy Wyświetlenie opisu obiektu (Display Object Description - DSPOBJD) do sprawdzenia, czy podpis jest tam nadal, i do ponownego podpisania obiektu, jeśli będzie to konieczne.

Uwaga: Aby sprawdzić, czy komenda Składowanie (Save) straciła podpis obiektu, należy odtworzyć obiekt w innej bibliotece, niż był składowany (na przykład w bibliotece QTEMP). Można następnie użyć komendy DSPOBJD do sprawdzenia, czy obiekt składowany utracił swój podpis.

- Komenda Zmiana programu (Change Program - CHGPGM) Komenda zmienia atrybuty programu bez potrzeby jego ponownej kompilacji. Można ją także wykorzystać do wymuszenia ponownego utworzenia programu, nawet jeśli określone atrybuty są identyczne jak bieżące.
- Komenda Zmiana programu usługowego (Change Service Program - CHGSRVPGM) Komenda zmienia atrybuty programu serwisowego bez potrzeby jego ponownej kompilacji. Można ją także wykorzystać do wymuszenia ponownego utworzenia programu serwisowego, nawet jeśli określone atrybuty są identyczne jak bieżące.
- Komenda Usuwanie zawartości zbioru składowania (Clear Save File - CLRSAVF) Komenda usuwa zawartość zbioru składowania, wszystkie istniejące rekordy zbioru składowania, zmniejszając ilość wykorzystywanej przez ten zbiór pamięci.
- Komenda Składowanie (Save - SAV) Komenda składuje kopię jednego lub większej liczby obiektów, z których można korzystać w zintegrowanym systemie plików. - podczas korzystania z tej komendy, jeśli dla parametru TGTRLS zostanie określona wartość wersji wcześniejsza niż V5R2M0, można utracić podpis z obiektów typu komenda (*CMD). Dzieje się tak dlatego, że w wersjach systemu wcześniejszych niż V5R2 nie było możliwości podpisywania komend.
- Komenda Składowanie biblioteki (Save Library - SAVLIB) Komenda umożliwia składowanie kopii jednej lub większej liczby bibliotek. Podczas korzystania z tej komendy, jeśli dla parametru TGTRLS zostanie podana wartość wersji wcześniejsza niż V5R2M0, można utracić podpis z obiektów typu komenda (*CMD). Dzieje się tak dlatego, że w wersjach systemu wcześniejszych niż V5R2 nie było możliwości podpisywania komend.
- Komenda Składowanie obiektu (Save Object - SAVOBJ) Komenda składuje kopię pojedynczego obiektu lub grupy obiektów położonych w tej samej bibliotece. Podczas korzystania z tej komendy, jeśli dla parametru TGTRLS zostanie podana wartość wersji wcześniejsza niż V5R2M0, można utracić podpis z obiektów typu komenda (*CMD). Dzieje się tak dlatego, że w wersjach systemu wcześniejszych niż V5R2 nie było możliwości podpisywania komend.

Pojęcia pokrewne

"Założenia związane ze składowaniem i odtwarzaniem podpisanych obiektów"
Ten temat zawiera informacje o wpływie podpisanych obiektów na sposób składowania i odtwarzania systemu, w którym działa system operacyjny i5/OS.

Informacje pokrewne

Wyszukiwarka wartości systemowych

Założenia związane ze składowaniem i odtwarzaniem podpisanych obiektów

Ten temat zawiera informacje o wpływie podpisanych obiektów na sposób składowania i odtwarzania systemu, w którym działa system operacyjny i5/OS.

Istnieje kilka wartości systemowych mających wpływ na operacje odtwarzania w systemie. Tylko jedna z tych wartości, a mianowicie wartość systemowa **verify object signatures during restore (QVFYOBJRST)** "Wartości systemowe i komendy wpływające na podpisane obiekty" na stronie 37 określa, w jaki sposób system przetwarza podpisane obiekty podczas ich odtwarzania. Ustawienia tej wartości systemowej pozwalają określić, jak proces odtwarzania obsługuje weryfikację obiektów bez podpisów lub obiektów, których podpisy są niepoprawne.

Niektóre komendy składowania i odtwarzania mają wpływ na podpisane obiekty lub określają, jak system obsługuje podpisane i niepodpisane obiekty podczas operacji składowania i odtwarzania. Aby lepiej zarządzać systemem i uniknąć potencjalnie możliwych problemów, należy być świadomym istnienia tych komend i ich wpływu na podpisane obiekty.

Następujące komendy mogą weryfikować podpisy na obiektach podczas operacji składowania i odtwarzania:
- Komenda Składowanie programu licencjonowanego (Save Licensed Program - SAVLICPGM)
- Komenda Odtworzenie (Restore - RST)
- Komenda Odtworzenie biblioteki (Restore Library - RSTLIB)
- Komenda Odtworzenie programu licencjonowanego (Restore Licensed Program - RSTLICPGM)
- Komenda Odtworzenie obiektu (Restore object - RSTOBJ)

Następujące komendy umożliwiają składowanie i odtwarzanie baz certyfikatów, które są istotne dla ochrony, gdyż zawierają certyfikaty używane do podpisywania obiektów i weryfikowania podpisów:
- Komenda Składowanie (Save - SAV)
- Komenda Składowanie danych ochrony (Save Security Data - SAVSECDTA)
- Komenda Składowanie systemu (Save System - SAVSYS)
- Komenda Odtworzenie (Restore - RST)
- Komenda Odtworzenie profili użytkowników (Restore User Profiles - RSTUSRPRF)

Niektóre komendy składowania, w zależności od użytych wartości parametrów, mogą usunąć podpis z obiektu umieszczanego na nośniku składowania, tym samym likwidując ochronę, jakiej dostarczał ten podpis. Na przykład *dowolna* operacja składowania, odnosząca się do obiektu typu komenda (*CMD), z docelową wersją systemu wcześniejszą niż V5R2M0, sprawia, że komenda zostanie składowana bez podpisu. Usunięcie podpisu może spowodować problemy z obiektem, z którego podpis został usunięty. W ostateczności nie będzie można zweryfikować źródła obiektu jako zaufanego lub zweryfikować podpisu w celu wykrycia zmian w obiekcie. Komend tych należy używać wyłącznie w odniesieniu do obiektów podpisanych utworzonych samodzielnie (w odróżnieniu do obiektów podpisanych otrzymanych z innych źródeł, na przykład od firmy IBM lub od dostawców).

Uwaga: Aby sprawdzić, czy komenda Składowanie (Save) straciła podpis obiektu, należy odtworzyć obiekt w innej bibliotece, niż był składowany (na przykład w bibliotece QTEMP). Można następnie użyć komendy DSPOBJD do sprawdzenia, czy obiekt składowany utracił swój podpis.

Należy mieć świadomość tej możliwości w przypadku następujących komend składowania (i ogólnie w przypadku komend składowania):
- Komenda Składowanie (Save - SAV)
- Komenda Składowanie biblioteki (Save Library - SAVLIB)
- Komenda Składowanie obiektu (Save Object - SAVOBJ)

Pojęcia pokrewne

“Wartości systemowe i komendy wpływające na podpisane obiekty” na stronie 37
Ten temat zawiera informacje o wartościach systemowych i komendach systemu operacyjnego i5/OS, które służą do zarządzania podpisanymi obiektami lub wpływają na nie po uruchomieniu.

Komendy sprawdzające kod w pod kątem integralności podpisu

Ten temat zawiera informacje o weryfikowaniu podpisów i sprawdzaniu integralności obiektów za pomocą komend systemu operacyjnego i5/OS.

Do weryfikowania podpisów na obiektach można użyć programu Digital Certificate Manager (DCM) lub funkcji API. Można także do tego celu użyć kilku komend. Korzysta się z nich w podobny sposób, jak z programu antywirusowego do określenia, czy wirus uszkodził jakieś pliki lub inne obiekty w systemie. Większość podpisów jest sprawdzana podczas odtwarzania lub instalowania obiektu na systemie, na przykład za pomocą komendy RSTLIB.

Do sprawdzenia podpisów na obiektach już znajdujących się w systemie, można użyć jednej z trzech komend. Jedna z nich, komenda Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG) jest przeznaczona specjalnie do weryfikacji podpisów obiektów. Na sprawdzanie podpisów przez każdą z tych komend ma wpływ parametr CHKSIG. Umożliwia on sprawdzenie wszystkich typów obiektów, ignorowanie wszystkich podpisów lub sprawdzanie tylko takich obiektów, które mają podpisy. Ostatnia opcja jest wartością domyślną tego parametru.

Komenda Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG)

Komenda Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG) umożliwia określenie, czy została naruszona integralność obiektów w systemie. Można za pomocą tej komendy sprawdzić naruszenie integralności obiektów należących do określonego profilu użytkownika, obiektów znajdujących się w miejscu o określonej nazwie ścieżki wszystkich obiektów w systemie. W protokole zostaje umieszczony wpis o naruszeniu integralności, jeśli nastąpiło jedno z następujących zdarzeń:

- Zostały zmienione atrybuty programu, komendy, obiektu modułu lub biblioteki.
- Podpis cyfrowy na obiekcie jest niepoprawny. Podpis jest zaszyfrowaną sumą danych w obiekcie, dlatego zakłada się, że podpis będzie zgodny i poprawny, jeśli dane w obiekcie podczas weryfikowania są zgodne z danymi w obiekcie w czasie jego podpisywania. Niepoprawny podpis jest określany przez porównanie zaszyfrowanej sumy tworzonej w momencie podpisywania obiektu i zaszyfrowanej sumy tworzonej w momencie weryfikowania podpisu. Proces weryfikacji podpisu porównuje te dwie wartości. Jeśli nie są identyczne, zawartość obiektu została zmieniona od czasu podpisania i zakłada się, że podpis jest niepoprawny.
- Obiekt ma niepoprawny atrybut domeny dla tego typu obiektu.

Jeśli komenda wykryje naruszenie integralności obiektu, to do protokołu bazy danych dodaje nazwę obiektu, nazwę biblioteki (lub ścieżkę), typ obiektu, właściciela obiektu i rodzaj błędu. Komenda tworzy także pozycję protokołu w różnych innych przypadkach, nawet jeśli nie są one związane z naruszeniem integralności. Na przykład komenda tworzy pozycję protokołu dla obiektów, które można podpisać, ale które nie mają podpisu cyfrowego, dla obiektów, których nie może sprawdzić i obiektów, które mają format, który wymagałby zmiany w bieżącej implementacji systemu (konwersji IMPI do RISC).

Wartość parametru CHKSIG określa, jak komenda obsługuje cyfrowe podpisy na obiektach. Jako wartość tego parametru można podać jedną z trzech wartości:

- Określenie wartości *SIGNED powoduje, że komenda sprawdza obiekty z podpisami cyfrowymi. Komenda tworzy pozycje protokołu dla każdego obiektu, który ma niepoprawny podpis. Jest to wartość domyślna.
- Określenie wartości *ALL powoduje, że komenda sprawdza wszystkie obiekty, które można podpisywać w celu określenia, czy są one podpisane. Komenda tworzy pozycje protokołu dla każdego obiektu, który można podpisywać, a który nie ma podpisu lub ma niepoprawny podpis.
- Określenie wartości *NONE powoduje, że komenda nie sprawdza podpisów cyfrowych obiektów.

Komenda Sprawdzenie opcji produktu (Check Product Option - CHKPRDOPT)

Komenda Sprawdzenie opcji produktu (Check Product Option - CHKPRDOPT) przedstawia różnice pomiędzy prawidłową a bieżącą strukturą oprogramowania. Na przykład pokaże błąd, jeśli z zainstalowanego produktu zostanie usunięty jakiś obiekt.

Wartość parametru CHKSIG określa, jak komenda obsługuje cyfrowe podpisy na obiektach. Jako wartość tego parametru można podać jedną z trzech wartości:

- Określenie wartości *SIGNED powoduje, że komenda sprawdza obiekty z podpisami cyfrowymi. Komenda weryfikuje podpisy na dowolnych podpisanych obiektach. Jeśli komenda wykryje, że podpis na obiekcie jest niepoprawny, wyśle komunikat do protokołu zadania i oznaczy produkt jako błędny. Jest to wartość domyślna.
- Określenie wartości *ALL powoduje, że komenda sprawdza wszystkie obiekty, które można podpisywać w celu określenia, czy są one podpisane, oraz weryfikuje te podpisy. Komenda wysyła komunikat do protokołu zadania dla każdego obiektu, który można podpisywać, a który nie ma podpisu, jednak nie oznaczy produktu jako błędnego. Jeśli komenda wykryje, że podpis na obiekcie jest niepoprawny, wyśle komunikat do protokołu zadania i oznaczy produkt jako błędny.

- Określenie wartości *NONE powoduje, że komenda nie sprawdza podpisów cyfrowych obiektów produktu.

Komenda Składowanie programu licencjonowanego (Save Licensed Program - SAVLICPGM)

Komenda Składowanie programu licencjonowanego (Save Licensed Program - SAVLICPGM) pozwala składować kopie obiektów tworzących program licencjonowany. Składuje go w takiej postaci, z której może zostać odtworzony komendą Odtworzenie programu licencjonowanego (Restore Licensed Program - RSTLICPGM).

Wartość parametru CHKSIG określa, jak komenda obsługuje cyfrowe podpisy na obiektach. Jako wartość tego parametru można podać jedną z trzech wartości:

- Określenie wartości *SIGNED powoduje, że komenda sprawdza obiekty z podpisami cyfrowymi. Komenda weryfikuje podpisy na dowolnych podpisanych obiektach, ale nie sprawdza niepodpisanych obiektów. Jeśli komenda wykryje, że podpis na obiekcie jest niepoprawny, wyśle komunikat do protokołu zadania, aby zidentyfikować produkt i składowanie nie powiedzie się. Jest to wartość domyślna.
- Określenie wartości *ALL powoduje, że komenda sprawdza wszystkie obiekty, które można podpisywać w celu określenia, czy są one podpisane, oraz weryfikuje te podpisy. Komenda wysyła komunikat do protokołu zadania dla każdego obiektu, który można podpisywać, a który nie ma podpisu; jednak proces składowania nie zostanie zakończony. Jeśli komenda wykryje, że podpis na obiekcie jest niepoprawny, wyśle komunikat do protokołu zadania i składowanie nie powiedzie się.
- Określenie wartości *NONE powoduje, że komenda nie sprawdza podpisów cyfrowych obiektów produktu.

Weryfikowanie integralności funkcji kontrolera kodu

Weryfikowanie integralności funkcji kontrolera kodu używanej do weryfikowania integralności systemu i5/OS.

Jeśli do weryfikowania integralności systemu ma być używana nowa funkcja weryfikacji integralności kontrolera kodu, użytkownik musi mieć uprawnienia specjalne *AUDIT.

Aby zweryfikować funkcję kontrolera kodu, uruchom funkcję API Check System (QydoCheckSystem), żeby ustalić czy któryś z kluczowych obiektów systemu operacyjnego został zmieniony od czasu kiedy go podpisano. Kiedy zostanie uruchomiona funkcja API, sprawdza ona kluczowe obiekty systemowe w tym: programy, programy usługowe, wybrane obiekty komend (*CMD) w bibliotece QSYS, w następujący sposób:

1. Sprawdza wszystkie programy (*PGM), do których odnosi się tabela punktów wejścia systemu.
2. Sprawdza wszystkie programy usługowe (*SRVPGM) w bibliotece QSYS i weryfikuje integralność funkcji API Verify Object.
3. Uruchamia funkcję API Verify Object (QydoVerifyObject), aby zweryfikować integralność komendy Odtworzenie obiektu (Restore Object - RSTOBJ), komendy Odtworzenie biblioteki (Restore Library - RSTLIB) i komendy Sprawdzenie integralności obiektu (Check Object Integrity - CHKOBJITG).
4. Używa komend RSTOBJ i RSTLIB na specjalnym zbiorze składowania (*SAV), aby upewnić się, że błędy są zgłaszane poprawnie. Brak komunikatów o błędach lub niewłaściwe komunikaty o błędach sygnalizują potencjalny problem.
5. Tworzy obiekt komendy (*CMD), który jest zaprojektowany tak, aby nie dał się zweryfikować poprawnie.
6. Uruchamia komendę CHKOBJITG i funkcję API Verify Object na tym specjalnym obiekcie komendy, aby zapewnić, że komenda CHKOBJITG i funkcja API Verify Object zgłaszają błędy poprawnie. Brak komunikatów o błędach lub niewłaściwe komunikaty o błędach sygnalizują potencjalny problem.
7. Weryfikuje podpis każdego modułu Licencjonowanego Kodu Wewnętrznego (kodu LIC) oraz sprawdza, czy zgłaszane są błędy w przypadku niepodpisanych lub nieprawidłowo podpisanych modułów LIC.

Pojęcia pokrewne

“Funkcja weryfikowania integralności kontrolera kodu” na stronie 6
Ten temat zawiera informacje o weryfikowaniu integralności kontrolera kodu, który służy do weryfikowania integralności systemów działających pod kontrolą systemu operacyjnego i5/OS.

Odsyłacze pokrewne

“Interpretowanie komunikatów o błędach weryfikacji kontrolera kodu” na stronie 45
Ten temat zawiera informacje o komunikatach generowanych przez funkcję weryfikowania integralności kontrolera kodu w systemie, w którym działa system operacyjny i5/OS. Ponadto opisano w nim sposób wykorzystania tych komunikatów do zapewnienia prawidłowego działania funkcji kontrolera kodu, a także potencjalne rozwiązania na potrzeby sytuacji, w których komunikat świadczy o możliwym uszkodzeniu funkcji lub kluczowych obiektów systemu operacyjnego.

Rozwiązywanie problemów z podpisanymi obiektami

Ten temat zawiera informacje o komendach i wartościach systemowych używanych w systemie operacyjnym i5/OS do pracy z podpisanymi obiektami. Ponadto opisano w nim wpływ podpisanych obiektów na procesy składowania i odtwarzania.

Podczas podpisywania obiektów i pracy z obiektami podpisanymi można napotkać błędy, które uniemożliwią zrealizowanie określonych zadań i celów. Wiele powszechnych błędów i problemów można podzielić na następujące kategorie:

Rozwiązywanie problemów związanych z błędami podpisywania obiektów

Ten temat zawiera informacje o rozwiązywaniu najczęściej występujących problemów związanych z podpisywaniem obiektów w systemie, w którym działa system operacyjny i5/OS.

Problem	Możliwe rozwiązanie
Podczas korzystania z funkcji API Sign Object do podpisywania obiektów z wersją docelową V4R5 lub wcześniejszą proces podpisywania kończy się błędem i obiekt pozostaje niepodpisany (komunikat o błędzie CPFB721).	System nie obsługuje podpisywania obiektów do wersji V5R1. Dla obiektów, które zwracają komunikat o błędzie CPFB721 trzeba w celu podpisania ponownie utworzyć te programy z docelową wersją systemu V5R1 lub nowszą.

Rozwiązywanie problemów związanych z błędami weryfikowania podpisów

Ten temat zawiera informacje o rozwiązywaniu najczęściej występujących problemów związanych z weryfikowaniem cyfrowych podpisów obiektów w systemie operacyjnym i5/OS.

Problem	Możliwe rozwiązanie
Proces odtwarzania nie powiódł się dla obiektów bez podpisów.	Jeśli brak podpisu nie jest problemem, należy sprawdzić, czy wartość systemowa QVFYOBJRST została ustawiona na 5. Wartość 5 określa, że niepodpisanych obiektów nie można odtwarzać. Należy zmienić tę wartość na 3 i spróbować ponowić odtwarzanie.
Proces odtwarzania dla podpisanych obiektów nie powiódł się.	Może się tak zdarzyć, jeśli baza certyfikatów *SIGNATUREVERIFICATION została przesłana do systemu, ale użyto programu DCM do zmiany hasła dla tej bazy. W takim przypadku certyfikaty zawarte w bazie nie mogą być użyte do weryfikowania podpisów na obiektach w czasie procesu odtwarzania. Należy użyć programu DCM i zmienić hasło bazy certyfikatów. Jeśli hasło jest nieznane, należy usunąć bazę certyfikatów, utworzyć ją ponownie i zmienić hasło za pomocą programu DCM.

Problem	Możliwe rozwiązanie
Podczas odtwarzania lub instalowania produktu pojawia się błąd, podpis nie daje się weryfikować.	Jeśli podpis nie daje się poprawnie weryfikować, może to oznaczać, że obiekt został zmieniony od czasu podpisania. Jeśli ważna jest integralność obiektu, nie należy zmieniać wartości systemowej QVFYOBJRST ani podejmować innych działań w celu odtworzenia podejrzanego obiektu. Może to bowiem spowodować obejście ochrony zapewnianej przez weryfikację podpisów i umożliwić przedostanie się do systemu obiektu, który może spowodować uszkodzenia. Należy natomiast skontaktować się z osobą podpisującą obiekt, aby określić, jaką akcję podjąć w celu rozwiązania problemu.

Interpretowanie komunikatów o błędach weryfikacji kontrolera kodu

Ten temat zawiera informacje o komunikatach generowanych przez funkcję weryfikowania integralności kontrolera kodu w systemie, w którym działa system operacyjny i5/OS. Ponadto opisano w nim sposób wykorzystania tych komunikatów do zapewnienia prawidłowego działania funkcji kontrolera kodu, a także potencjalne rozwiązania na potrzeby sytuacji, w których komunikat świadczy o możliwym uszkodzeniu funkcji lub kluczowych obiektów systemu operacyjnego.

Poniższa tabela zawiera listę komunikatów, które funkcja weryfikacji kontrolera kodu generuje podczas przetwarzania. Ta tabela nie jest pełną listą komunikatów, które mogą być wyświetlane. Tabela zawiera jedynie komunikaty sygnalizujące, że weryfikacja kontrolera kodu w pełni się powiodła lub wykryła poważny problem. Szczegółowa lista komunikatów o błędach znajduje się w dokumentacji funkcji API Check System (QydoCheckSystem).

Pewna liczba komunikatów generowanych podczas przetwarzania przez funkcję weryfikacji kontrolera kodu ma charakter informacyjny i nie została uwzględniona w tabeli. Więcej informacji o tym, jak działa proces weryfikacji kontrolera kodu zawiera temat Weryfikowanie integralności funkcji kontrolera kodu.

Tabela 1. Komunikaty o błędach weryfikacji kontrolera kodu

Komunikat o błędzie	Możliwe rozwiązanie
CPFB729	Sygnalizuje, że proces weryfikacji integralności kontrolera kodu nie został zakończony zgodnie z oczekiwaniem. To niepowodzenie może być spowodowane przez różnorakie problemy. Aby ustalić dokładny rodzaj niepowodzenia oraz możliwą przyczynę jego zaistnienia, należy zapoznać się z protokołem zadania zawierającym bardziej szczegółowe komunikaty o błędach. Jeśli ustalono, że nie powiodła się weryfikacja integralności kluczowych obiektów systemu operacyjnego, może to oznaczać, że obiekt został zmieniony od momentu podpisania, kiedy dostarczono system operacyjny. Dla zapewnienia integralności systemu może zachodzić konieczność reinstalacji systemu operacyjnego.
Podczas przeglądania protokołu zadania napotkano komunikaty takie jak: CPFB723, CPD37A1 lub CPD37A0 dla następujących obiektów: • Obiekty programów (*PGM): – QYDONOSIG w bibliotece QTEMP – QYDOBADSIG w bibliotece QTEMP • Obiekty komend (*CMD): – QYDOBADSIG w bibliotece QTEMP – SIGNOFF w bibliotece QTEMP	Sygnalizuje, że test specjalnego zbioru obiektów, którego funkcja weryfikacji kontrolera kodu używa do testowania integralności, nie powiódł się zgodnie z oczekiwaniem. To niepowodzenie sygnalizuje, że komenda RSTOBJ, komenda RSTLIB, komenda CHKOBJITG oraz funkcja API Verify Object zgłaszają błędy prawidłowo. Nie są konieczne żadne dalsze działania.

Tabela 1. Komunikaty o błędach weryfikacji kontrolera kodu (kontynuacja)

Komunikat o błędzie	Możliwe rozwiązanie
CPFB723 dla wszystkich obiektów, które nie zostały wymienione powyżej.	Sygnalizuje, że weryfikacja podpisu kluczowego obiektu systemu operacyjnego nie powiodła się. To niepowodzenie może oznaczać, że obiekt został zmieniony od momentu podpisania, kiedy system operacyjny został dostarczony. Dla zapewnienia integralności systemu może zachodzić konieczność reinstalacji systemu operacyjnego.
CPFB722 dla wszystkich obiektów, które nie zostały wymienione powyżej.	Sygnalizuje, że kluczowy obiekt systemu operacyjnego jest nie podpisany, kiedy podpis jest oczekiwany. Ten brak podpisu może sygnalizować, że obiekt został zmieniony od momentu podpisania, kiedy system operacyjny został dostarczony. Dla zapewnienia integralności systemu może zachodzić konieczność reinstalacji systemu operacyjnego.
CPFB72A dla wszystkich obiektów, które nie zostały wymienione powyżej.	Sygnalizuje, że sprawdzenie integralności kluczowego obiektu systemu operacyjnego nie powiodło się. To niepowodzenie może oznaczać, że obiekt został zmieniony od momentu podpisania, kiedy system operacyjny został dostarczony. Dla zapewnienia integralności systemu może zachodzić konieczność reinstalacji systemu operacyjnego.

Zawsze, kiedy zachodzi potrzeba reinstalacji kodu, który weryfikuje integralność funkcji kontrolera kodu, należy go uzyskiwać ze znanego, dobrego źródła. Na przykład można załadować nośnik instalacji użyty do zainstalowania wersji bieżącej. Aby odtworzyć funkcję weryfikacji kontrolera kodu, wykonaj następujące czynności używając wiersza komend systemu i5/OS:

1. Uruchom komendę QSYS/DLTPGM QSYS/QYDOCHKS. Ta komenda usuwa funkcję API Check System (OPM, QYDOCHKS; ILE, QydoCheckSystem).
2. Uruchom komendę QSYS/DLTSRVPGM QSYS/QYDOCHK1. Ta komenda usuwa program usługowy kontrolera kodu z funkcją API Check System (OPM, QYDOCHKS; ILE, QydoCheckSystem).
3. Uruchom komendę QSYS/DLTF QSYS/QYDOCHKF. Ta komenda usuwa zbiór składowania zawierający obiekty, których funkcja kontrolera kodu używa do testowania pod kątem błędnych podpisów lub braku podpisów.
4. Uruchom komendę QSYS/RSTOBJ OBJ(QYDOCHK*) SAVLIB(QSYS) DEV(OPT01) OBJTYPE(*ALL) OPTFILE('Q5722SS1/Q5200M_/Q00/Q90'). Ta komenda przywraca wszystkie niezbędne dla funkcji weryfikacji kontrolera kodu obiekty z załadowanego nośnika instalacji.

Zadania pokrewne

"Weryfikowanie integralności funkcji kontrolera kodu" na stronie 43
Weryfikowanie integralności funkcji kontrolera kodu używanej do weryfikowania integralności systemu i5/OS.

Informacje pokrewne dotyczące podpisywania obiektów i weryfikowania podpisów

Informacje pokrewne dotyczące kolekcji tematów Podpisywanie obiektów i weryfikowanie podpisów znajdują się w serwisach WWW i plikach PDF dokumentacji technicznej IBM (Redbooks). Wszystkie pliki PDF można wyświetlić lub wydrukować.

Podpisywanie obiektów i weryfikacja podpisów są stosunkowo nową technologią ochrony. Oto krótka lista innych pomocnych źródeł informacji dla osób zainteresowanych szerszym zrozumieniem tych technologii i ich działania:

- **Serwis WWW VeriSign Help Desk** Serwis WWW firmy VeriSign zawiera dużo informacji dotyczących certyfikatów cyfrowych (na przykład o podpisywaniu obiektów) oraz szeregu innych zagadnień związanych z bezpieczeństwem w Internecie.

- **IBM eServer iSeries Wired Network Security: i5/OS V5R1 DCM and Cryptographic Enhancements**

  SG24-6168  Ta dokumentacja techniczna IBM (Redbooks) dotyczy głównie rozszerzeń funkcji zabezpieczania sieci w wersji V5R1. Obejmuje ona wiele tematów, w tym opis możliwości podpisywania obiektów, programu Digital Certificate Manager (DCM) itd.

Licencja na kod oraz Informacje dotyczące kodu

IBM udziela niewyłącznej licencji na prawa autorskie, stosowanej przy używaniu wszelkich przykładowych kodów programów, na podstawie których można wygenerować podobne funkcje dostosowane do indywidualnych wymagań.

Z ZASTRZEŻENIEM GWARANCJI WYNIKAJĄCYCH Z BEZWZGLĘDNIE OBOWIĄZUJĄCYCH PRZEPISÓW PRAWA, IBM, PROGRAMIŚCI ANI DOSTAWCY IBM NIE UDZIELAJĄ NA NINIEJSZY PROGRAM ANI W ZAKRESIE EWENTUALNEGO WSPARCIA TECHNICZNEGO ŻADNYCH GWARANCJI, W TYM TAKŻE RĘKOJMI, NIE USTALAJĄ ŻADNYCH WARUNKÓW, WYRAŹNYCH CZY DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI CZY WARUNKÓW PRZYDATNOŚCI HANDLOWEJ, PRZYDATNOŚCI DO OKREŚLONEGO CZY NIENARUSZANIA PRAW STRON TRZECICH.

W ŻADNYCH OKOLICZNOŚCIACH IBM, ANI TEŻ PROGRAMIŚCI CZY DOSTAWCY PROGRAMÓW IBM, NIE PONOSZĄ ODPOWIEDZIALNOŚCI ZA PONIŻSZE SZKODY, NAWET JEŚLI ZOSTALI POINFORMOWANI O MOŻLIWOŚCI ICH WYSTĄPIENIA:

1. UTRATA LUB USZKODZENIE DANYCH;
2. SZKODY BEZPOŚREDNIE, SZCZEGÓLNE, UBOCZNE, POŚREDNIE ORAZ SZKODY, KTÓRYCH NIE MOŻNA BYŁO PRZEWIDZIEĆ PRZY ZAWIERANIU UMOWY, ANI TEŻ
3. UTRATA ZYSKÓW, KONTAKTÓW HANDLOWYCH, PRZYCHODÓW, REPUTACJI (GOODWILL) LUB PRZEWIDYWANYCH OSZCZĘDNOŚCI.

USTAWODAWSTWA NIEKTÓRYCH KRAJÓW NIE DOPUSZCZAJĄ WYŁĄCZENIA CZY OGRANICZENIA ODPOWIEDZIALNOŚCI ZA SZKODY BEZPOŚREDNIE, UBOCZNE LUB SZKODY, KTÓRYCH NIE MOŻNA BYŁO PRZEWIDZIEĆ PRZY ZAWIERANIU UMOWY, W ZWIĄZKU Z CZYM W ODNIESIENIU DO NIEKTÓRYCH KLIENTÓW POWYŻSZE WYŁĄCZENIE LUB OGRANICZENIE (TAK W CAŁOŚCI JAK I W CZĘŚCI) MOŻE NIE MIEĆ ZASTOSOWANIA.

Dodatek. Uwagi

Niniejsza publikacja została przygotowana z myślą o produktach i usługach oferowanych w Stanach Zjednoczonych.

IBM może nie oferować w innych krajach produktów, usług lub opcji, omawianych w tej publikacji. Informacje o produktach i usługach dostępnych w danym kraju można uzyskać od lokalnego przedstawiciela IBM. Odwołanie do produktu, programu lub usługi IBM nie oznacza, że można użyć wyłącznie tego produktu, programu lub usługi. Zamiast nich można zastosować ich odpowiednik funkcjonalny pod warunkiem, że nie narusza to praw własności intelektualnej IBM. Jednakże cała odpowiedzialność za ocenę przydatności i sprawdzenie działania produktu, programu lub usługi pochodzących od producenta innego niż IBM spoczywa na użytkowniku.

IBM może posiadać patenty lub złożone wnioski patentowe na towary i usługi, o których mowa w niniejszej publikacji. Przedstawienie niniejszej publikacji nie daje żadnych uprawnień licencyjnych do tychże patentów. Pisemne zapytania w sprawie licencji można przesyłać na adres:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

Zapytania w sprawie licencji na informacje dotyczące zestawów znaków dwubajtowych (DBCS) należy kierować do lokalnych działów własności intelektualnej IBM (IBM Intellectual Property Department) lub zgłaszać na piśmie pod adresem:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokio 106-0032, Japonia

Poniższy akapit nie obowiązuje w Wielkiej Brytanii, a także w innych krajach, w których jego treść pozostaje w sprzeczności z przepisami prawa miejscowego: INTERNATIONAL BUSINESS MACHINES CORPORATION DOSTARCZA TĘ PUBLIKACJĘ W STANIE, W JAKIM SIĘ ZNAJDUJE “AS IS” BEZ UDZIELANIA JAKICHKOLWIEK GWARANCJI (W TYM TAKŻE RĘKOJMI), WYRAŹNYCH LUB DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI PRZYDATNOŚCI HANDLOWEJ, PRZYDATNOŚCI DO OKREŚLONEGO CELU ORAZ GWARANCJI, ŻE PUBLIKACJA NIE NARUSZA PRAW STRON TRZECICH. Ustawodawstwa niektórych krajów nie dopuszczają zastrzeżeń dotyczących gwarancji wyraźnych lub domniemanych w odniesieniu do pewnych transakcji; w takiej sytuacji powyższe zdanie nie ma zastosowania.

Informacje zawarte w niniejszej publikacji mogą zawierać nieścisłości techniczne lub błędy drukarskie. Informacje te są okresowo aktualizowane, a zmiany te zostaną uwzględnione w kolejnych wydaniach tej publikacji. IBM zastrzega sobie prawo do wprowadzania ulepszeń i/lub zmian w produktach i/lub programach opisanych w tej publikacji w dowolnym czasie, bez wcześniejszego powiadomienia.

Wszelkie wzmianki w tej publikacji na temat stron internetowych innych firm zostały wprowadzone wyłącznie dla wygody użytkowników i w żadnym wypadku nie stanowią zachęty do ich odwiedzania. Materiały dostępne na tych stronach nie są częścią materiałów opracowanych dla tego produktu IBM, a użytkownik korzysta z nich na własną odpowiedzialność.

IBM ma prawo do korzystania i rozpowszechniania informacji przysłanych przez użytkownika w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich autora.

Licencjobiorcy tego programu, którzy chcieliby uzyskać informacje na temat programu w celu: (i) wdrożenia wymiany informacji między niezależnie utworzonymi programami i innymi programami (łącznie z tym opisywanym) oraz (ii) wspólnego wykorzystywania wymienianych informacji, powinni skontaktować się z:

IBM Corporation
Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
USA

Informacje takie mogą być udostępnione, o ile spełnione zostaną odpowiednie warunki, w tym, w niektórych przypadkach, uiszczenie odpowiedniej opłaty.

Licencjonowany program opisany w niniejszej publikacji oraz wszystkie inne licencjonowane materiały dostępne dla tego programu są dostarczane przez IBM na warunkach określonych w Umowie IBM z Klientem, Międzynarodowej Umowie Licencyjnej IBM na Program, Umowie Licencyjnej IBM na Kod Maszynowy lub w innych podobnych umowach zawartych między IBM i użytkownikami.

Wszelkie dane dotyczące wydajności zostały zebrane w kontrolowanym środowisku. W związku z tym rezultaty uzyskane w innych środowiskach operacyjnych mogą się znacząco różnić. Niektóre pomiary mogły być dokonywane na systemach będących w fazie rozwoju i nie ma gwarancji, że pomiary te wykonane na ogólnie dostępnych systemach dadzą takie same wyniki. Niektóre z pomiarów mogły być estymowane przez ekstrapolację. Rzeczywiste wyniki mogą być inne. Użytkownicy powinni we własnym zakresie sprawdzić odpowiednie dane dla ich środowiska.

Informacje dotyczące produktów firm innych niż IBM pochodzą od dostawców tych produktów, z opublikowanych przez nich zapowiedzi lub innych powszechnie dostępnych źródeł. Firma IBM nie testowała tych produktów i nie może potwierdzić dokładności pomiarów wydajności, kompatybilności ani żadnych innych danych związanych z tymi produktami. Pytania dotyczące możliwości produktów firm innych niż IBM należy kierować do dostawców tych produktów.

Wszelkie stwierdzenia dotyczące przyszłych kierunków rozwoju i zamierzeń IBM mogą zostać zmienione lub wycofane bez powiadomienia.

Wszelkie ceny podawane przez IBM są propozycjami cen detalicznych; ceny te są aktualne i podlegają zmianom bez wcześniejszego powiadomienia. Ceny podawane przez dealerów mogą być inne.

Niniejsza informacja służy jedynie do celów planowania. Informacja ta podlega zmianom do chwili, gdy produkty, których ona dotyczy, staną się dostępne.

Publikacja ta zawiera przykładowe dane i raporty używane w codziennych operacjach działalności gospodarczej. W celu kompleksowego ich zilustrowania, podane przykłady zawierają nazwiska osób prywatnych, nazwy przedsiębiorstw oraz nazwy produktów. Wszystkie te nazwy/nazwiska są fikcyjne i jakiekolwiek podobieństwo do istniejących nazw/nazwisk i adresów jest całkowicie przypadkowe.

LICENCJA W ZAKRESIE PRAW AUTORSKICH:

Niniejsza publikacja zawiera przykładowe aplikacje w kodzie źródłowym, ilustrujące techniki programowania w różnych systemach operacyjnych. Użytkownik może kopiować, modyfikować i dystrybuować te programy przykładowe w dowolnej formie bez uiszczania opłat na rzecz IBM, w celu projektowania, używania, sprzedaży lub dystrybucji aplikacji zgodnych z aplikacyjnym interfejsem programowym dla tego systemu operacyjnego, dla którego napisane zostały programy przykładowe. Programy przykładowe nie zostały gruntownie przetestowane. IBM nie może zatem gwarantować ani sugerować niezawodności, użyteczności i funkcjonalności tych programów.

Z UWZGLĘDNIENIEM WSZELKICH BEZWZGLĘDNIE OBOWIĄZUJĄCYCH GWARANCJI, KTÓRYCH NIE WOLNO WYKLUCZYĆ, IBM, PROGRAMIŚCI IBM ORAZ DOSTAWCY NIE UDZIELAJĄ W ZAKRESIE TEGO PROGRAMU CZY EWENTUALNEGO WSPARCIA TECHNICZNEGO ŻADNYCH GWARANCJI (W TYM TAKŻE RĘKOJMI), ANI NIE USTALAJĄ WARUNKÓW, WYRAŹNYCH CZY DOMNIEMANYCH, A W

SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI CZY WARUNKÓW PRZYDATNOŚCI HANDLOWEJ, PRZYDATNOŚCI DO OKREŚLONEGO CELU CZY NIENARUSZANIA PRAW STRON TRZECICH.

W ŻADNYM PRZYPADKU IBM, PROGRAMIŚCI IBM ANI DOSTAWCY NIE PONOSZĄ ODPOWIEDZIALNOŚCI ZA PONIŻSZE STRATY LUB SZKODY, NAWET JEŚLI BYLIBY POINFORMOWANI O MOŻLIWOŚCI ICH WYSTĄPIENIA:

1. UTRATA LUB USZKODZENIE DANYCH;
2. SZKODY SZCZEGÓLNE, UBOCZNE LUB POŚREDNIE, A TAKŻE SZKODY, KTÓRYCH NIE MOŻNA BYŁO PRZEWIDZIEĆ PRZY ZAWIERANIU UMOWY; ORAZ
3. UTRATA ZYSKÓW, KONTAKTÓW HANDLOWYCH, PRZYCHODÓW, REPUTACJI (GOODWILL) LUB PRZEWIDYWANYCH OSZCZĘDNOŚCI.

USTAWODAWSTWA NIEKTÓRYCH KRAJÓW NIE DOPUSZCZAJĄ WYŁĄCZENIA ANI OGRANICZENIA ODPOWIEDZIALNOŚCI ZA SZKODY UBOCZNE LUB SZKODY, KTÓRYCH NIE MOŻNA BYŁO PRZEWIDZIEĆ PRZY ZAWIERANIU UMOWY, W ZWIĄZKU Z CZYM W ODNIESIENIU DO NIEKTÓRYCH KLIENTÓW POWYŻSZE WYŁĄCZENIE LUB OGRANICZENIE MOŻE NIE MIEĆ ZASTOSOWANIA.

Każda kopia programu przykładowego lub jakikolwiek jego fragment, jak też jakiekolwiek prace pochodne muszą zawierać następujące uwagi dotyczące praw autorskich:

W przypadku przeglądania niniejszych informacji w formie elektronicznej, zdjęcia i kolorowe ilustracje mogą nie być wyświetlane.

Znaki towarowe

Następujące nazwy są znakami towarowymi International Business Machines Corporation w Stanach Zjednoczonych i/lub w innych krajach:

Adobe eServer
i5/OS
IBM
iSeries
OS/400
Redbooks
system i
xSeries

Adobe, logo Adobe, PostScript oraz logo PostScript są zastrzeżonymi znakami towarowymi lub znakami towarowymi firmy Adobe Systems Incorporated w Stanach Zjednoczonych i/lub w innych krajach.

Microsoft, Windows, Windows NT oraz logo Windows są znakami towarowymi Microsoft Corporation w Stanach Zjednoczonych i/lub w innych krajach.

Java oraz wszystkie znaki towarowe dotyczące języka Java są znakami towarowymi Sun Microsystems, Inc. w Stanach Zjednoczonych i/lub w innych krajach.

Linux jest znakiem towarowym Linusa Torvaldsa w Stanach Zjednoczonych i/lub w innych krajach.

UNIX jest zastrzeżonym znakiem towarowym Open Group w Stanach Zjednoczonych i w innych krajach.

Nazwy innych przedsiębiorstw, produktów i usług mogą być znakami towarowymi lub znakami usług innych podmiotów.

Warunki

Zezwolenie na korzystanie z tych publikacji jest przyznawane na poniższych warunkach.

Użytek osobisty: Użytkownik ma prawo kopiować te publikacje do własnego, niekomercyjnego użytku pod warunkiem zachowania wszelkich uwag dotyczących praw własności. Użytkownik nie ma prawa dystrybuować ani wyświetlać tych publikacji czy ich części, ani też wykonywać na ich podstawie prac pochodnych bez wyraźnej zgody IBM.

Użytek służbowy: Użytkownik ma prawo kopiować te publikacje, dystrybuować je i wyświetlać wyłącznie w ramach przedsiębiorstwa Użytkownika pod warunkiem zachowania wszelkich uwag dotyczących praw własności. Użytkownik nie ma prawa wykonywać na podstawie tych publikacji ani ich fragmentów prac pochodnych, kopiować ich, dystrybuować ani wyświetlać poza przedsiębiorstwem Użytkownika bez wyraźnej zgody IBM.

Z wyjątkiem zezwoleń wyraźnie udzielonych w niniejszym dokumencie, nie udziela się jakichkolwiek innych zezwoleń, licencji ani praw, wyraźnych czy domniemanych, odnoszących się do tych publikacji czy jakichkolwiek informacji, danych, oprogramowania lub innej własności intelektualnej, o których mowa w niniejszym dokumencie.

IBM zastrzega sobie prawo do anulowania zezwolenia przyznanego w niniejszym dokumencie w każdej sytuacji, gdy, według uznania IBM, korzystanie z tych publikacji jest szkodliwe dla IBM lub jeśli IBM uzna, że warunki niniejszego dokumentu nie są przestrzegane.

Użytkownik ma prawo pobierać, eksportować lub reeksportować niniejsze informacje pod warunkiem zachowania bezwzględnej i pełnej zgodności z obowiązującym prawem i przepisami, w tym ze wszelkimi prawami i przepisami eksportowymi Stanów Zjednoczonych.

IBM NIE UDZIELA JAKICHKOLWIEK GWARANCJI, W TYM TAKŻE RĘKOJMI, DOTYCZĄCYCH TREŚCI TYCH PUBLIKACJI. PUBLIKACJE TE SĄ DOSTARCZANE W STANIE, W JAKIM SIĘ ZNAJDUJĄ (″AS IS″) BEZ UDZIELANIA JAKICHKOLWIEK GWARANCJI, W TYM TAKŻE RĘKOJMI, WYRAŹNYCH CZY DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI PRZYDATNOŚCI HANDLOWEJ, PRZYDATNOŚCI DO OKREŚLONEGO CELU ORAZ NIENARUSZANIA PRAW STRON TRZECICH.

Drukowane w USA