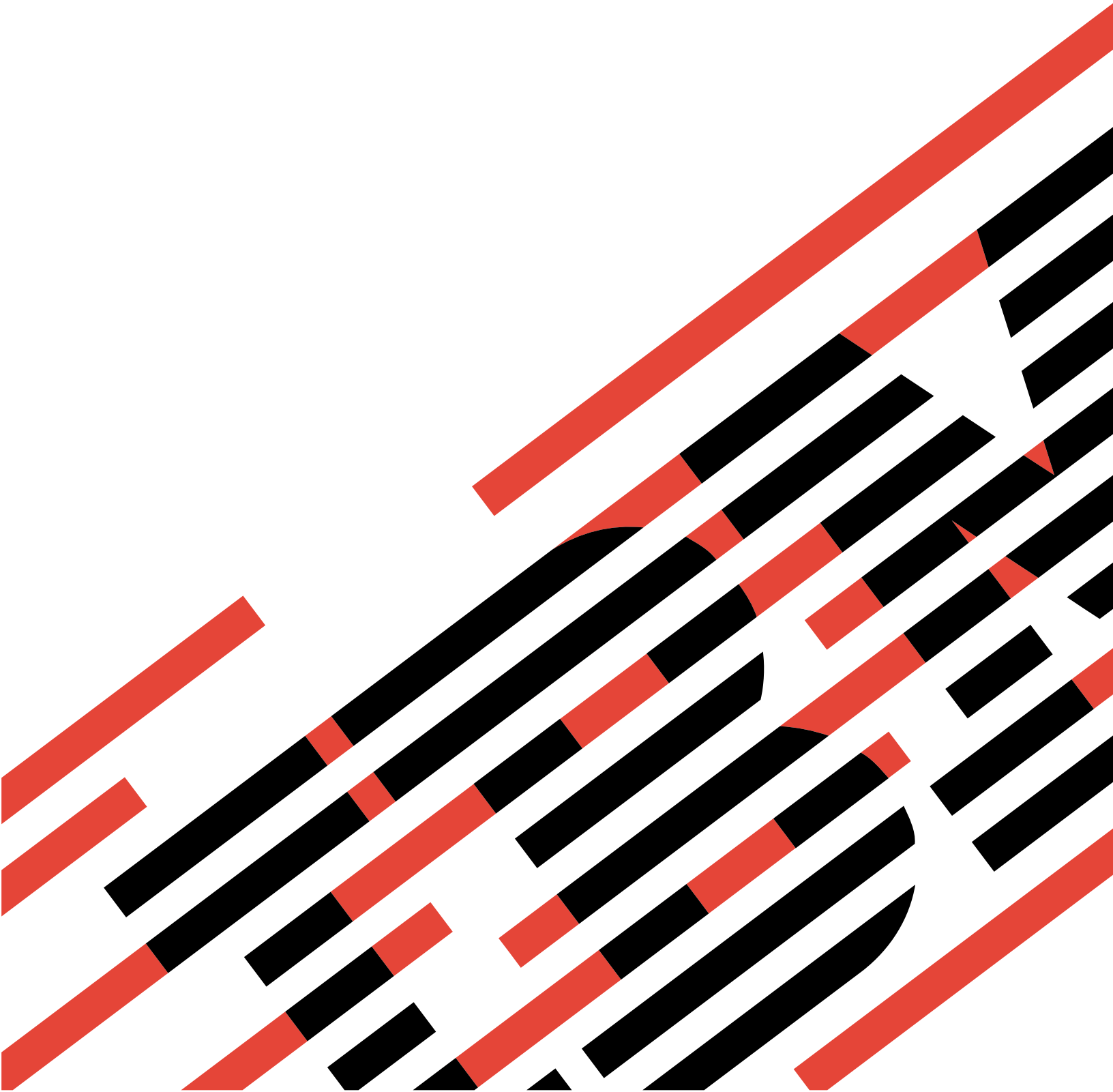
IBM

eserver

iSeries

오브젝트 서명 및 서명 확인



IBM

@server

iSeries

오브젝트 서명 및 서명 확인

목차

오브젝트 서명 및 서명 확인

오브젝트 서명 및 서명 확인은 다양한 iSeries™ 오브젝트의 무결성을 확인하기 위해 사용할 수 있는 보안 기능입니다. 디지털 인증의 개인 키를 사용하여 오브젝트에 서명하고, 해당하는 공용 키가 포함된 인증서를 사용하여 디지털 서명을 확인할 수 있습니다. 디지털 서명을 사용하면 서명하는 오브젝트의 시간 및 내용의 무결성이 보장됩니다. 서명은 신뢰성과 권한을 의미합니다. 서명은 권한 없는 변경을 감지하거나 원본을 확인하는 데 사용할 수 있습니다. 오브젝트에 서명하면 오브젝트의 소스를 식별하고 오브젝트에 대한 변경사항을 감지하는 방법을 제공합니다. 오브젝트의 서명을 확인하면 서명 이후에 오브젝트 내용이 변경되었는지 여부를 판별할 수 있습니다. 또한 오브젝트 원본의 신뢰성을 보장하기 위해 서명의 소스를 확인할 수도 있습니다.

다음을 사용하여 iSeries 오브젝트 서명 및 서명 확인을 구현할 수 있습니다.

- 오브젝트에 서명하고 오브젝트의 서명을 프로그래밍 방식으로 확인하기 위한 API.
- 오브젝트에 서명하고 오브젝트 서명을 보거나 확인하기 위한 디지털 인증 관리자.
- 다른 시스템에서 사용할 수 있도록 패키지를 분배하는 작업의 일부로 오브젝트에 서명하는 iSeries Navigator 중앙 관리.
- CHKOBJITG(오브젝트 무결성 검사) 명령과 같이 서명을 확인하기 위한 CL 명령.

오브젝트 서명 방법과 오브젝트 서명으로 현재 보안 정책을 개선할 수 있는 방법에 대한 자세한 내용은 다음 주제를 검토하십시오.

V5R2의 새로운 사항
이 릴리스의 문서 변경사항 및 새로운 iSeries 오브젝트 서명 및 서명 확인 기능에 대해 알려면 이 정보를 사용하십시오.

이 주제 인쇄
전체 주제를 PDF 파일로 인쇄하려면 이 정보를 사용하십시오.

오브젝트 서명 시나리오
iSeries 오브젝트 서명 및 서명 확인 기능을 사용하기 위한 일반적인 상황을 설명하는 시나리오를 검토하려면 이 정보를 사용하십시오. 또한, 각 시나리오는 설명된 시나리오를 구현하기 위해 수행해야 하는 구성 타스크를 제공합니다.

오브젝트 서명 개념
디지털 서명과 오브젝트 서명 및 서명 확인 프로세스 작업에 대해 자세히 알려면 이 개념 및 참조 정보를 사용하십시오.

오브젝트 서명 및 서명 확인의 전제조건
오브젝트 서명 및 서명 확인을 위한 기타 계획 고려사항 및 구성 전제조건에 대해 알려면 이 정보를 사용하십시오.

서명된 오브젝트 관리
서명된 오브젝트에 대한작업을 수행하기 위해 사용할 수 있는 iSeries 명령 및 시스템 값 또는 서명된 오브젝트가 백업 및 회복 프로세스에 미치는 영향에 대해 알려면 이 정보를 사용하십시오.

오브젝트 서명 및 서명 확인 문제 해결
오브젝트에 서명하고 서명을 확인할 때 발생할 수 있는 문제점 및 오류를 해결하는 방법을 알려면 이 정보를 사용하십시오.

오브젝트 서명 및 서명 확인에 대한 관련 정보
오브젝트 서명 및 오브젝트 서명 확인에 대해 자세히 알기 위해 다른 자원에 대한 링크를 찾으려면 이 정보를 사용하십시오.

V5R2의 새로운 사항

iSeries의 오브젝트 서명 및 서명 확인 기능은 V5R1에서 처음 소개되었습니다. V5R2에서는 일부 새로운 기능과 확장 기능을 사용할 수 있습니다.

추가된 오브젝트 서명 및 서명 확인 기능에는 다음이 포함됩니다.

- **iSeries Navigator 중앙 관리 오브젝트 서명 기능**
  이제는 중앙 관리 제품 정의 마법사를 사용하여 iSeries 종료점 시스템에 분배하기 위해 패키지하는 오브젝트에 서명할 수 있습니다.
- **명령(*CMD) 오브젝트 서명**
  이제 명령(*CMD) 오브젝트에 서명할 수 있습니다. 전체 *CMD 오브젝트에 서명할지 또는 *CMD 오브젝트의 핵심 구성요소에만 서명할지 선택할 수 있습니다.
- **새로운 서명 및 확인 API**
  세 개의 새로운 API를 사용하여 프로그래밍 방식으로 OS/400® 서명 및 확인 기능에 대한 확장 기능을 이용할 수 있습니다.
  - Sign Buffer(QYDOSGNB, QydoSignBuffer) API
    이 API를 사용하면 로컬 시스템이 버퍼에 디지털 서명하여 버퍼를 신뢰할 수 있는지 증명할 수 있습니다. 버퍼에 서명하면 시스템은 디지털 서명을 API 호출자에게 리턴합니다. 예를 들어, 이 API를 사용하여 XML 파일의 일부를 서명하고 XML 파일의 다른 부분에 이 서명을 저장할 수 있습니다. 또는 데이터베이스 파일 레코드를 버퍼로 읽어들이고 API를 사용하여 이 레코드에 서명할 수 있습니다.
  - Verify Buffer(QYDOVFYB, QydoVerifyBuffer) API
    이 API를 사용하면 로컬 시스템에서 이전에 서명한 버퍼의 디지털 서명을 확인할 수 있습니다.
  - Add Verifier(QYDOADDV, QydoAddVerifier) API
    이 API는 인증서를 시스템의 *SIGNATUREVERIFICATION 인증서 저장소에 추가합니다. 그러면 시스템이 추가된 인증서를 사용하여 인증서가 작성한 오브젝트의 서명을 확인할 수 있습니다. 서명을 확인하면 시스템이 서명된 오브젝트의 무결성을 확인하여 서명 후 오브젝트가 변경되지 않았음을 보장할 수 있습니다. 인증서 저장소가 없는 경우에는 인증서 추가 시 이 API가 인증서 저장소를 작성합니다.

주: 보안상의 이유로 이 API는 사용자가 CA(Certificate Authority) 인증서를 *SIGNATUREVERIFICATION 인증서 저장소에 넣도록 허용하지 않습니다. CA 인증서를 인증서 저장소에 추가한 경우 시스템은 해당 CA를 신뢰할 수 있는 인증서의 소스로 간주합니다. 따라서 시스템은 이 CA에서 발행한 인증서를 신뢰할 수 있는 소스에서 나온 인증서로서 처리합니다. 그러므로 CA 인증서를 인증서 저장소에 넣기 위해 API를 사용하여 설치 나감 프로그램을 작성할 수 없습니다. CA 인증서를 인증서 저장소에 추가하여 시스템이 신뢰하는 CA를 수동으로 확실하게 제어하려면 디지털 인증 관리자를 사용해야 합니다. 디지털 인증 관리자를 사용하면 시스템이 관리자가 신뢰하지 않은 소스에서 인증서를 가져올 가능성을 방지할 수 있습니다.

누구도 사용자의 허락 없이 API를 사용하여 확인 인증서를 *SIGNATUREVERIFICATION 인증서 저장소에 추가하지 못하도록 하려면 시스템에서 이 API가 작동하지 않도록 설정하십시오. SST(System Service Tool)를 사용하여 보안 관련 시스템 값을 변경할 수 없도록 설정하면 이렇게 할 수 있습니다.

이전에는 iSeries 오브젝트 서명 및 서명 확인 기능에 대한 정보를 디지털 인증 관리 Information Center 주제의 일부에서 사용할 수 있었습니다. 그러나 이제는 오브젝트 서명 및 서명 확인에 대해 추가적인 방법을 사용할 수 있습니다. 따라서 이 새로운 Information Center 주제를 사용하면 기능 사용에 대한 집중화된 정보를 통해 오브젝트 서명과 서명 확인 기능을 더 쉽게 사용할 수 있습니다. 이 주제에서는 보안 정책을 보완하기 위해 이 기능을 사용하는 시기 및 방법을 판별할 수 있도록 시나리오와 같은 개선되고 보다 완전한 정보를 제공합니다.

이 주제에 대한 새로운 정보나 향상된 정보에는 다음이 포함됩니다.

- 보안 정책을 보완하기 위해 오브젝트 서명과 서명 확인 기능을 사용하는 방법을 판별할 수 있도록 도와주는 시나리오.
- 시스템에서 서명된 오브젝트를 관리하는 데 사용할 수 있는 명령과 시스템 값을 설명하는 새로운 섹션.
- 오브젝트 서명과 서명 확인에 대한 계획 및 기타 개념적인 정보를 설명하는 새로운 섹션.

이 릴리스의 새로운 사항이나 변경된 사항에 대한 기타 정보를 찾아보려면 사용자 메모 를 참조하십시오.

이 주제 인쇄

PDF 버전을 보거나 다운로드하려면 오브젝트 서명 및 서명 확인 (파일 크기 350kb 또는 약 44페이지)을 선택하십시오.

워크스테이션에 PDF를 저장하여 보거나 인쇄하려면 다음을 수행하십시오.

1. 브라우저에서 PDF를 여십시오(위의 링크를 클릭).
2. 브라우저 메뉴에서 **파일**을 클릭하십시오.
3. **다른 이름으로 저장**을 클릭하십시오.
4. PDF를 저장할 디렉토리로 이동하십시오.

5. **저장**을 클릭하십시오.

PDF를 보거나 인쇄하기 위해 Adobe Acrobat Reader가 필요한 경우 Adobe 웹 사이트 (www.adobe.com/prodindex/acrobat/readstep.html)  에서 사본을 다운로드할 수 있습니다.

오브젝트 서명 시나리오

iSeries 서버는 오브젝트 서명과 오브젝트 서명 확인을 위한 여러 가지 방법을 제공합니다. 오브젝트 서명 선택과 서명된 오브젝트에 대한 작업을 수행하는 방법은 사용자의 업무, 보안 요구사항 및 목표에 따라 다릅니다. 오브젝트의 무결성이 손상되지 않았는지 확인하기 위해 시스템의 오브젝트 서명을 확인하려는 경우가 있을 수 있습니다. 또한 다른 사람에게 분배하는 오브젝트에 서명하려는 경우도 있습니다. 오브젝트에 서명하면 다른 사용자가 오브젝트의 원본을 식별하고 오브젝트의 무결성을 검사할 수 있습니다.

어떤 방법을 사용할 것인지는 여러 요소에 따라 달라집니다. 이 주제에서 제공하는 시나리오는 일반적인 업무 상황 안에서 보다 일반적인 오브젝트 서명 및 서명 확인 목표를 설명합니다. 또한 설명한 대로 시나리오를 구현하기 위해 수행해야 하는 타스크와 전제조건을 제공합니다. 이 시나리오를 검토하면 사용자의 업무 및 보안 요구사항에 가장 알맞은 iSeries 오브젝트 서명 기능 사용 방법을 판별할 수 있습니다.

시나리오: 오브젝트에 서명하고 오브젝트 서명을 확인하기 위해 디지털 인증 관리자 사용
이 시나리오에서는 공용 웹 서버의 보안이 취약한 어플리케이션 오브젝트에 서명하려는 회사에 대해 설명합니다. 이 회사는 이 오브젝트에 대해 권한 없는 변경사항이 있을 경우 이를 보다 쉽게 판별하기를 원합니다. 회사의 업무 요구사항과 보안 목표에 따라 이 시나리오에서는 오브젝트에 서명하고 오브젝트 서명을 확인하기 위한 기본 방법으로 DCM(Digital Certificate Manager)을 사용하는 방법에 대해 설명합니다.

시나리오: 오브젝트에 서명하고 서명을 확인하기 위해 API 사용
이 시나리오에서는 자사에서 판매하는 어플리케이션에 프로그래밍 방식으로 서명하려는 어플리케이션 개발 회사에 대해 설명합니다. 이 회사는 고객에게 해당 어플리케이션이 자사에서 제공되었음을 확인시켜주고, 어플리케이션을 설치할 때 권한 없는 변경사항을 감지할 수 있는 방법을 제공하려고 합니다. 회사의 업무 요구사항과 보안 목표에 따라 이 시나리오에서는 Sign Object API와 Add Verifier API를 사용하여 오브젝트에 서명 및 서명을 확인하는 방법에 대해 설명합니다.

시나리오: 오브젝트에 서명하기 위해 중앙 관리 사용
이 시나리오에서는 자사에서 패키지하여 복수 iSeries 서버에 분배하는 오브젝트에 서명하려는 회사에 대해 설명합니다. 회사의 업무 요구사항과 보안 목표에 따라 이 시나리오에서는 iSeries Navigator의 중앙 관리 기능을 사용하여 다른 iSeries 서버에 분배하는 오브젝트를 패키지하고 서명하는 방법에 대해 설명합니다.

시나리오: 오브젝트에 서명하고 서명을 확인하기 위해 DCM 사용

상황

MyCo., Inc.의 iSeries 관리자로서, 사용자는 회사에 있는 두 개의 iSeries 서버를 관리할 책임이 있습니다. 이 iSeries 서버 중 한 대가 회사의 공용 웹 사이트를 제공합니다. 이 공용 웹 사이트의 내용을 개발하고 파일과 프로그램 오브젝트를 테스트한 후 공용 웹 서버에 전송하기 위해 사용자는 회사의 내부 프로덕션 iSeries 서버를 사용합니다.

회사의 공용 웹 서버에서는 일반적인 회사 정보 웹 사이트를 제공합니다. 이 웹 사이트에서는 제품을 등록하거나 제품 정보, 제품 갱신 공지, 제품 분배 위치 등을 요청하기 위해 고객이 작성하는 다양한 양식도 제공합니다. 이러한 양식을 제공하는 cgi-bin 프로그램은 보안에 취약하기 때문에 이 프로그램에 대한 보안 문제를 보완해야 한다는 것을 알고 있습니다. 따라서 이 프로그램 오브젝트의 무결성을 점검하여 해당 오브젝트에 권한 없는 변경사항이 작성된 시기를 감지하려고 합니다. 결국, 이러한 보안 목표를 달성하기 위해 이 오브젝트에 디지털 서명을 하기로 결정했습니다.

OS/400 오브젝트 서명 기능에 대한 연구를 통해 오브젝트에 서명하고 오브젝트 서명을 확인하기 위해 사용할 수 있는 방법에는 여러 가지가 있다는 것을 알았습니다. 몇 대의 iSeries 서버만 관리하면 되고 오브젝트에 자주 서명하지 않아도 되기 때문에 디지털 인증 관리자(DCM)를 사용하여 이러한 타스크를 수행하기로 결정했습니다. 또한 로컬 CA(Certificate Authority)를 작성하고 개인 인증서를 사용하여 오브젝트에 서명하기로 결정했습니다. 오브젝트 서명 시 로컬 CA에서 발행한 개인 인증서를 사용하면 잘 알려진 공공 인증 기관의 인증서를 구매하지 않아도 되기 때문에 보안 기술 사용 비용을 제한할 수 있습니다.

이 예제는 적은 수의 iSeries 서버에 있는 오브젝트에 서명할 경우 오브젝트 서명 설정 및 사용에 관련된 단계에 대한 유용한 개요로 사용할 수 있습니다.

시나리오 장점

이 시나리오에는 다음과 같은 장점이 있습니다.

- 오브젝트 서명은 보안 문제가 있는 오브젝트의 무결성을 검사하고 서명 후 오브젝트의 변경 여부를 더 쉽게 확인할 수 있는 방법을 제공합니다. 따라서 어플리케이션과 다른 시스템 문제를 추적하기 위해 향후 수행할 문제 해결 작업의 일부를 줄일 수 있습니다.
- 오브젝트에 서명하고 오브젝트 서명을 확인하기 위해 DCM의 GUI(Graphical User Interface)를 사용하면 회사의 모든 사용자들이 이 타스크를 빠르고 쉽게 수행할 수 있습니다.
- 오브젝트에 서명하고 오브젝트 서명을 확인하기 위해 DCM을 사용하면 보안 정책의 일부인 오브젝트 서명을 이해하고 사용하는 데 걸리는 시간을 줄일 수 있습니다.
- 오브젝트 서명 시 로컬 CA(Certificate Authority)에서 발행한 인증서를 사용하면 오브젝트 서명을 구현하는 데 드는 비용을 줄일 수 있습니다.

목표

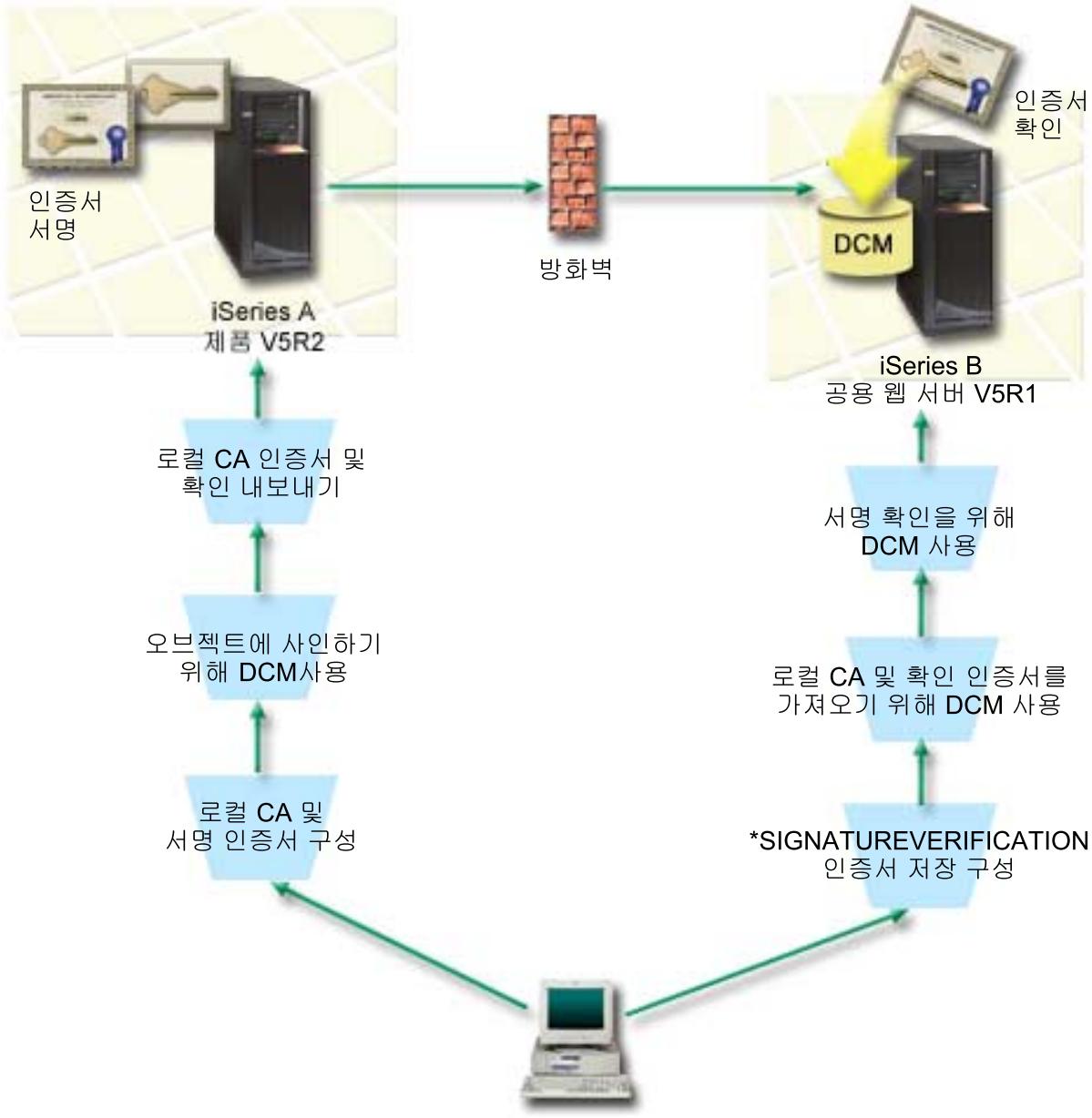
이 시나리오에서 사용자는 회사의 공용 iSeries 서버에서 보안 문제가 있는 오브젝트(예: 양식을 생성하는 cgi-bin 프로그램)에 디지털로 서명하려고 합니다. MyCo, Inc.의 시스템 관리자로서, DCM(Digital Certificate Manager)을 사용하여 이 오브젝트에 서명하고 오브젝트의 서명을 확인하려고 합니다.

이 시나리오의 목표는 다음과 같습니다.

- 어플리케이션 서명 비용을 줄이기 위해 회사 어플리케이션과 공용 웹 서버(iSeries B)의 다른 보안 문제 오브젝트를 로컬 CA의 인증서로 서명해야 합니다.
- 시스템 관리자와 다른 지정한 사용자가 iSeries 서버의 디지털 서명을 쉽게 확인해서 회사의 서명한 오브젝트의 소스와 신뢰성을 확인할 수 있어야 합니다. 이를 수행하려면 모든 iSeries 서버의 *SIGNATUREVERIFICATION 인증서 저장소에 회사의 서명 확인 인증서와 로컬 CA(Certificate Authority) 인증서의 사본이 있어야 합니다.
- 회사 어플리케이션과 다른 오브젝트의 서명을 확인하여 iSeries 관리자와 기타 사용자는 오브젝트에 서명한 이후 오브젝트 내용이 변경되었는지 확인할 수 있습니다.
- 시스템 관리자는 오브젝트에 서명할 때 DCM을 사용해야 하고, 시스템 관리자 및 기타 사용자는 오브젝트 서명 확인을 위해 DCM을 사용할 수 있어야 합니다.

세부사항

다음 그림은 이 시나리오를 구현하기 위한 오브젝트 서명 및 서명 확인 프로세스를 보여줍니다.



이 그림은 이 시나리오와 관련된 다음 사항을 설명합니다.

iSeries A

- iSeries A에서는 OS/400 버전 5 릴리스 2(V5R2)를 실행합니다.
- iSeries A는 회사의 내부 프로덕션 서버이고 공용 iSeries 웹 서버(iSeries B)의 개발 플랫폼입니다.
- iSeries A에는 iSeries용 Cryptographic Access Provider 128비트(5722-AC3)가 설치되어 있습니다.
- iSeries A에는 디지털 인증 관리자(OS/400 옵션 34)와 IBM® HTTP 서버(5722-DG1)가 설치 및 구성되어 있습니다.

- iSeries A는 로컬 CA(Certificate Authority) 역할을 하며 오브젝트 서명 인증서가 이 시스템에 상주합니다.
- iSeries A는 DCM을 사용하여 오브젝트에 서명합니다. 이는 회사 공용 어플리케이션 및 기타 오브젝트의 1차 오브젝트 서명 시스템입니다.
- iSeries A는 서명 확인이 가능하도록 구성되었습니다.

iSeries B

- iSeries B에서는 OS/400 버전 5 릴리스 1(V5R1)을 실행합니다.
- iSeries B는 회사 방화벽 외부에 있는 회사의 외부 공용 웹 서버입니다.
- iSeries B에는 Cryptographic Access Provider 128비트(5722-AC3)가 설치되어 있습니다.
- iSeries B에는 디지털 인증 관리자(OS/400 옵션 34)와 IBM HTTP 서버(5722-DG1)가 설치 및 구성되어 있습니다.
- iSeries B는 로컬 CA 및 iSeries B 서명 오브젝트 모두를 작동하지 않습니다.
- iSeries B는 *SIGNATUREVERIFICATION 인증서 저장소를 작성하고 필요한 확인 및 로컬 CA 인증서를 가져오기 위해 DCM을 사용하여 서명을 확인할 수 있도록 구성되었습니다.
- DCM을 사용하여 오브젝트의 서명을 확인합니다.

전제조건 및 가정

이 시나리오는 다음 전제조건과 가정에 따라 달라집니다.

1. 모든 iSeries 서버가 DCM(Digital Certificate Manager)을 설치하고 사용하기 위한 요구사항을 충족시킵니다.
2. 이전에 다른 사용자가 iSeries 서버에 DCM을 구성하거나 사용하지 않았습니다.
3. 모든 iSeries 서버에 최상위 수준의 Cryptographic Access Provider 128비트 사용권 프로그램(5722-AC3)이 설치되어 있습니다.
4. 모든 시나리오 iSeries 서버에서 복원 중 오브젝트 서명 확인(QVFYOBJRST) 시스템 값의 디폴트 설정은 3이고, 이 설정이 변경되지 않았습니다. 디폴트 설정을 사용하면 사용자가 서명된 오브젝트를 복원할 때 서버에서 오브젝트 서명을 확인할 수 있습니다.
5. iSeries A의 시스템 관리자는 오브젝트를 서명할 수 있는 *ALLOBJ 특수 권한이 있거나 오브젝트 서명 어플리케이션에 대해 사용자 프로파일에 권한이 부여되어야 합니다.
6. DCM에 인증서 저장소를 작성하는 시스템 관리자나 다른 사용자는 *SECADM 및 *ALLOBJ 특수 권한이 있어야 합니다.
7. 시스템 관리자나 다른 모든 iSeries 서버의 사용자가 오브젝트 서명을 확인하려면 *AUDIT 특수 권한이 있어야 합니다.

타스크 단계

이 시나리오를 구현하려면 완료해야 할 두 가지 타스크 세트가 있습니다. 첫 번째 타스크 세트를 사용하면 iSeries A를 로컬 CA(Certificate Authority)로 구성하고 오브젝트 서명을 확인할 수 있습니다. 두 번째 타스크 세트를 사용하면 iSeries A가 작성한 오브젝트 서명을 확인할 수 있도록 iSeries B를 구성할 수 있습니다.

iSeries A 타스크 단계

이 시나리오에서 설명한 대로 개인 로컬 CA를 작성하고, 오브젝트를 서명하고, 오브젝트 서명을 확인하려면 iSeries A에서 이 타스크를 완료해야 합니다.

1. 필요한 모든 iSeries 제품을 설치 및 구성하려면 모든 전제조건 단계를 완료하십시오.
2. DCM(Digital Certificate Manager)을 사용하여 오브젝트 서명 인증서를 발행하기 위한 로컬 CA(Certificate Authority)를 작성하십시오.
3. DCM을 사용하여 어플리케이션 정의를 작성하십시오.
4. DCM을 사용하여 오브젝트 서명 어플리케이션 정의에 인증서를 할당하십시오.
5. DCM을 사용하여 cgi-bin 프로그램 오브젝트에 서명하십시오.
6. DCM을 사용하여 오브젝트 서명을 확인하기 위해 다른 시스템에서 사용해야 할 인증서를 내보내십시오. 로컬 CA 인증서의 사본과 오브젝트 서명 인증서의 사본을 서명 확인 인증서로 파일에 내보내야 합니다.
7. 회사의 공용 iSeries 서버(iSeries B)로 인증서 파일을 전송하여 모든 사용자가 iSeries A에서 작성한 서명을 확인할 수 있게 하십시오.

iSeries B 타스크 단계

이 시나리오의 공용 웹 서버(iSeries B)로 전송한 서명한 오브젝트를 복원할 경우 서명된 오브젝트를 전송하기 전에 iSeries B에서 이 서명 확인 구성 타스크를 완료해야 합니다. 서명된 오브젝트를 공용 웹 서버에 복원할 때 서명을 확인하려면 서명 확인 구성을 완료해야 합니다.

iSeries B에서 이 시나리오에서 설명한 대로 오브젝트의 서명을 확인하려면 다음 타스크를 완료해야 합니다.

8. DCM(Digital Certificate Manager)을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 작성하십시오.
9. DCM을 사용하여 로컬 CA 인증서와 서명 확인 인증서를 가져오십시오.
10. DCM을 사용하여 전송된 오브젝트의 서명을 확인하십시오.

구성 세부사항

디지털 인증 관리자를 구성 및 사용하기 위한 다음 타스크 단계를 완료하여 이 시나리오에서 설명한 대로 오브젝트에 서명하십시오.

1단계: 모든 전제조건 단계 완료

이 시나리오 구현을 위한 특정한 구성 타스크를 수행하려면 필요한 모든 iSeries 제품을 설치 및 구성하는 전제조건 타스크를 모두 완료해야 합니다.

2단계: 개인 오브젝트 서명 인증서를 발행할 로컬 CA 작성

DCM(Digital Certificate Manager)을 사용하여 로컬 CA(Certificate Authority)를 작성할 경우 프로세스에서 일련의 양식을 완료해야 합니다. 이 양식은 SSL(Secure Sockets Layer), 오브젝트 서명 및 서명 확인을 위한 디지털 인증서를 사용하기 위해 완료해야 하는 타스크와 CA를 작성하기 위한 프로세스를 안내해줍니다. 이 시나리오에서는 SSL용 인증서를 구성하지 않아도 되지만 시스템이 오브젝트를 서명할 수 있도록 구성하는 타스크에서 모든 양식을 완료해야 합니다.

DCM을 사용하여 로컬 CA를 작성하고 작동시키려면 다음 단계를 수행하십시오.

1. DCM을 시작하십시오.
2. DCM의 탐색 프레임에서 **CA(Certificate Authority) 작성**을 선택하여 일련의 양식을 표시하십시오.

   주: 안내된 이 타스크에서 특정 양식을 완료하는 방법에 대해 의문이 있으면 페이지 위쪽에 있는 물음표 (**?**) 버튼을 선택하여 온라인 도움말에 액세스하십시오.
3. 안내된 이 타스크의 모든 양식을 완료하십시오. 이 타스크를 수행할 때 다음을 수행해야 합니다.
   a. 로컬 CA에 식별 정보를 제공하십시오.
   b. 브라우저에 로컬 CA 인증서를 설치하여 소프트웨어에서 로컬 CA를 인식하고 로컬 CA에서 발행하는 인증서를 확인할 수 있게 하십시오.
   c. 로컬 CA에 대한 정책 데이터를 지정하십시오.
   d. 새로운 로컬 CA를 사용하여 어플리케이션에서 SSL 연결에 사용할 수 있는 서버나 클라이언트 인증서를 발행하십시오.

      주: 이 시나리오에서는 이 인증서를 사용하지 않지만 인증서를 작성해야 로컬 CA를 사용하여 필요한 오브젝트 서명 인증서를 발행할 수 있습니다. 이 인증서를 작성하지 않고 타스크를 취소한 경우 오브젝트 서명 인증서와 오브젝트 서명 인증서가 저장되는 *OBJECTSIGNING 인증서 저장소를 각각 작성해야 합니다.
   e. SSL 연결을 위해 서버 인증서나 클라이언트 인증서를 사용할 수 있는 어플리케이션을 선택하십시오.

      주: 이 시나리오 대로 진행하려면 어플리케이션을 선택하지 말고 **계속**을 클릭하여 다음 양식을 표시하십시오.
   f. 새로운 로컬 CA를 사용하여 어플리케이션에서 오브젝트를 디지털 서명하기 위해 사용할 수 있는 오브젝트 서명 인증서를 발행하십시오. 이 서브타스크는 *OBJECTSIGNING 인증서 저장소를 작성합니다. 이것이 바로 오브젝트 서명 인증서를 관리하기 위해 사용하는 인증서 저장소입니다.
   g. 로컬 CA를 신뢰하는 어플리케이션을 선택하십시오.

      주: 이 시나리오 대로 진행하려면 어플리케이션을 선택하지 말고 **계속**을 클릭하여 타스크를 완료하십시오.

이제 로컬 CA와 오브젝트 서명 인증서를 작성했습니다. 오브젝트를 서명하려면 인증서를 사용할 수 있도록 오브젝트 서명 어플리케이션을 정의해야 합니다.

3단계: 오브젝트 서명 어플리케이션 정의 작성

오브젝트 서명 인증서를 작성한 후에는 DCM(Digital Certificate Manager)을 사용하여 오브젝트에 서명하는 데 사용할 수 있는 오브젝트 서명 어플리케이션을 정의해야 합니다. 어플리케이션 정의에서 반드시 실제 어플리케이션을 참조할 필요는 없습니다. 그러나 작성한 어플리케이션 정의에서 서명하려고 하는 오브젝트의 유형이나 그룹을 설명해야 합니다. 어플리케이션 ID를 인증서와 연관시켜 서명 프로세스를 가능하게 하려면 정의가 필요합니다.

DCM을 사용하여 오브젝트 서명 어플리케이션 정의를 작성하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **인증서 저장소 선택**을 클릭한 다음 ***OBJECTSIGNING**을 열어야 할 인증서 저장소로 선택하십시오.
2. 인증서 저장소 및 암호 페이지가 표시되면 인증서 저장소를 작성할 때 인증서 저장소에 지정한 암호를 제공한 다음 **계속**을 클릭하십시오.
3. 탐색 프레임에서 **어플리케이션 관리**를 선택하여 타스크 리스트를 표시하십시오.
4. 타스크 리스트에서 **어플리케이션 추가**를 선택하여 어플리케이션을 정의하기 위한 양식을 표시하십시오.
5. 양식을 완료하고 **추가**를 클릭하십시오.

이제 오브젝트 서명 인증서를 작성한 어플리케이션에 할당해야 합니다.

4단계: 오브젝트 서명 어플리케이션 정의에 인증서 할당

오브젝트 서명 어플리케이션에 인증서를 할당하려면 다음 단계를 수행하십시오.

1. DCM 탐색 프레임에서 **인증서 관리**를 선택하여 타스크 리스트를 표시하십시오.
2. 타스크 리스트에서 **인증서 할당**을 선택하여 현재 인증서 저장소의 인증서 리스트를 표시하십시오.
3. 리스트에서 인증서를 선택하고 **어플리케이션에 할당**을 클릭하여 현재 인증서 저장소의 어플리케이션 정의 리스트를 표시하십시오.
4. 리스트에서 하나 이상의 어플리케이션을 선택하고 **계속**을 클릭하십시오. 인증서 할당을 확인하는 메시지 페이지가 표시되거나 문제가 발생한 경우 오류 정보를 제공하는 메세지 페이지가 표시됩니다.

이 타스크를 완료하면 이제 DCM을 사용하여 회사의 공용 웹 서버(iSeries B)에서 사용할 프로그램 오브젝트를 서명할 수 있습니다.

5단계: 프로그램 오브젝트에 서명

DCM을 사용하여 회사의 공용 웹 서버(iSeries B)에서 사용할 프로그램 오브젝트에 서명하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **인증서 저장소 선택**을 클릭한 다음 ***OBJECTSIGNING**을 열 인증서 저장소로 선택하십시오.
2. *OBJECTSIGNING 인증서 저장소에 대한 암호를 입력한 다음 **계속**을 클릭하십시오.
3. 탐색 프레임의 화면을 정리한 후 **서명 가능한 오브젝트 관리**를 선택하여 타스크 리스트를 표시하십시오.

4. 타스크 리스트에서 **오브젝트 서명**을 선택하여 오브젝트를 서명하기 위해 사용할 수 있는 어플리케이션 정의 리스트를 표시하십시오.
5. 이전 단계에서 정의한 어플리케이션을 선택한 다음 **오브젝트 서명**을 클릭하십시오. 서명할 오브젝트의 위치를 지정할 수 있게 해주는 양식이 표시됩니다.
6. 제공된 필드에서, 오브젝트의 완전한 경로 및 파일명 또는 서명할 오브젝트의 디렉토리를 입력한 다음 **계속**을 클릭하십시오. 또는 디렉토리 위치를 입력하고 **찾아보기**를 클릭하여 디렉토리의 내용을 보면서 서명할 오브젝트를 선택하십시오.

   주: 오브젝트 이름 앞에는 슬래시가 있어야 합니다. 그렇지 않으면 오류가 발생합니다. 서명할 디렉토리의 일부를 나타내기 위해 특정한 와일드카드 문자를 사용할 수도 있습니다. 이런 와일드카드 문자에는 *임의의 수의 문자*를 지정하는 별표(*****)와 *하나의 문자*를 지정하는 물음표(**?**)가 있습니다. 예를 들어, 특정 디렉토리의 모든 오브젝트에 서명하려면 `/mydirectory/*;`를 입력할 수 있고, 특정 라이브러리의 모든 프로그램에 서명하려면 `/QSYS.LIB/QGPL.LIB/*.PGM`을 입력할 수 있습니다. 이 와일드카드는 경로명의 마지막 부분에만 사용할 수 있습니다. 예를 들어, `/mydirectory*/filename`을 입력하면 오류 메세지가 표시됩니다. 라이브러리 리스트나 디렉토리 내용을 보기 위해 찾아보기 기능을 사용하려면 **찾아보기**를 클릭하기 전에 경로명의 일부로 와일드카드를 입력해야 합니다.
7. 선택한 오브젝트를 서명하기 위해 사용할 처리 옵션을 선택한 다음 **계속**을 클릭하십시오.

   주: 작업 결과를 기다리기로 선택한 경우 결과 파일이 직접 브라우저로 표시됩니다. 현재 작업의 결과가 결과 파일 끝에 추가됩니다. 따라서 파일에는 현재 작업의 결과 외에도 이전 작업의 결과가 포함될 수 있습니다. 파일의 날짜 필드를 사용하여 파일의 어떤 행이 현재 작업에 적용되었는지 확인할 수 있습니다. 날짜 필드의 형식은 YYYYMMDD입니다. 파일의 첫 번째 필드는 메시지 ID(오브젝트 처리 중에 오류가 발생한 경우)이거나 날짜 필드(작업이 처리된 날짜를 가리킴)일 수 있습니다.
8. 오브젝트 서명 작업의 결과를 저장하는 데 사용할 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하십시오. 또는 디렉토리 위치를 입력하고 **찾아보기**를 클릭하여 디렉토리의 내용을 보면서 작업 결과를 저장할 파일을 선택하십시오. 오브젝트에 서명하기 위해 작업이 제출되었음을 나타내는 메시지가 표시됩니다. 작업 결과를 보려면 작업 기록부의 **QOBJSGNBAT** 작업을 참조하십시오.

사용자나 다른 사용자가 서명을 확인할 수 있게 하려면 필요한 인증서를 파일로 내보낸 다음, 인증서 파일을 iSeries B로 전송해야 합니다. 또한 서명된 프로그램 오브젝트를 iSeries B로 전송하기 전에 iSeries B에서 모든 서명 확인 구성 타스크를 완료해야 합니다. 서명 확인 구성을 완료해야 iSeries B에서 서명된 오브젝트를 복원할 때 서명을 확인할 수 있습니다.

6단계: iSeries B에서 서명을 확인할 수 있도록 인증서 내보내기

내용의 무결성을 보호하기 위한 오브젝트 서명의 경우, 사용자나 기타 사용자에게는 서명의 신뢰성을 확인할 수 있는 수단이 있어야 합니다. 오브젝트에 서명한 동일한 시스템(iSeries A)에서 오브젝트 서명을 확인하려면 DCM을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 작성해야 합니다. 이 인증서 저장소에는 오브젝트 서명 인증서의 사본과 서명 인증서를 발행한 CA의 CA 인증서 사본이 모두 들어 있어야 합니다.

다른 사용자가 서명을 확인할 수 있게 하려면 그들에게 오브젝트를 서명한 인증서의 사본을 제공해야 합니다. 로컬 CA(Certificate Authority)를 사용하여 인증서를 발행할 경우 로컬 CA 인증서의 사본도 제공해야 합니다.

DCM을 사용하여 오브젝트를 서명한 같은 시스템(이 시나리오에서는 iSeries A)에서 서명을 확인하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **신규 인증서 저장소 작성**을 선택한 다음 ***SIGNATUREVERIFICATION**을 새로 작성할 인증서 저장소로 선택하십시오.
2. **예**를 선택하여 기존의 오브젝트 서명 인증서를 신규 인증 저장소에 서명 확인 인증서로 복사하십시오.
3. 새로운 인증서 저장소의 암호를 지정하고 **계속**을 클릭하여 인증서 저장소를 작성하십시오. 오브젝트에 서명하기 위해 사용한 같은 시스템에서 이제는 DCM을 사용하여 오브젝트 서명을 확인할 수 있습니다.

DCM을 사용하여 로컬 CA 인증서의 사본과 오브젝트 서명 인증서의 사본을 서명 확인 인증서로 내보냄으로써 다른 시스템(iSeries B)에서 오브젝트 서명을 확인할 수 있게 하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **인증서 관리**를 선택한 다음 **인증서 내보내기** 타스크를 선택하십시오.
2. **CA(Certificate Authority)**를 선택하고 **계속**을 클릭하여 내보낼 수 있는 CA 인증서 리스트를 표시하십시오.
3. 리스트에서 이전에 작성한 로컬 CA 인증서를 선택하고 **내보내기**를 클릭하십시오.
4. 내보내기 대상으로 **파일**을 지정하고 **계속**을 클릭하십시오.
5. 내보낸 로컬 CA 인증서의 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하여 인증서를 내보내십시오.
6. **확인**을 클릭하여 내보내기 확인 페이지에서 나가십시오. 이제는 오브젝트 서명 인증서 사본을 내보낼 수 있습니다.
7. **인증서 내보내기** 타스크를 다시 선택하십시오.
8. **오브젝트 서명**을 선택하여 내보낼 수 있는 오브젝트 서명 인증서 리스트를 표시하십시오.
9. 리스트에서 해당하는 오브젝트 서명 인증서를 선택한 다음 **내보내기**를 클릭하십시오.
10. 대상으로 **파일, 서명 확인 인증서로**를 선택한 다음 **계속**을 클릭하십시오.
11. 내보낸 서명 확인 인증서의 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하여 인증서를 내보내십시오.

이제 인증서로 작성한 서명을 확인할 iSeries 종료점 시스템으로 이 파일을 전송할 수 있습니다.

7단계: 인증서 파일을 회사 공용 서버 iSeries B로 전송

서명한 오브젝트를 확인할 수 있도록 인증서 파일을 구성하려면 iSeries A에서 작성한 인증서 파일을 이 시나리오에서 회사의 공용 웹 서버인 iSeries B로 전송해야 합니다. 여러 다른 방법을 사용하여 인증서 파일을 전송할 수 있습니다. 예를 들어, FTP(File Transfer Protocol)나 중앙 관리 패키지 분배를 사용하여 파일을 전송할 수 있습니다.

8단계: 서명 확인 타스크: *SIGNATUREVERIFICATION 인증서 저장소 작성

회사의 공용 웹 서버인 iSeries B에서 오브젝트 서명을 확인하려면 iSeries B는 해당 서명 확인 인증서 사본을 *SIGNATUREVERIFICATION 인증서 저장소에 가지고 있어야 합니다. 로컬에서 발행한 인증서를 사용하여 오브젝트에 서명하기 때문에 이 인증서 저장소에는 로컬 CA 인증서의 사본도 있어야 합니다.

*SIGNATUREVERIFICATION 인증서 저장소를 작성하려면 다음 단계를 수행하십시오.

1. DCM을 시작하십시오.
2. DCM(Digital Certificate Manager) 탐색 프레임에서 **신규 인증서 저장소 작성**을 선택한 다음 ***SIGNATUREVERIFICATION**을 새로 작성할 인증서 저장소로 선택하십시오.

   주: DCM을 사용하는 중에 특정 양식을 완료하는 방법에 의문이 있으면 페이지 위쪽에 있는 물음표(**?**)를 선택하여 온라인 도움말에 액세스하십시오.
3. 새로운 인증서 저장소의 암호를 지정하고 **계속**을 클릭하여 인증서 저장소를 작성하십시오. 이제는 인증서를 저장소로 가져온 후 이 인증서를 사용하여 오브젝트 서명을 확인할 수 있습니다.

9단계: 서명 확인 타스크: 인증서 가져오기

오브젝트의 서명을 확인하려면 *SIGNATUREVERIFICATION 저장소에 서명 확인 인증서의 사본이 있어야 합니다. 서명 인증서가 개인 인증서일 경우 이 인증서 저장소에는 서명 인증서를 발행한 로컬 CA(Certificate Authority) 인증서의 사본도 있어야 합니다. 이 시나리오에서는 두 개의 인증서를 파일로 내보냈으며 해당 파일을 각각의 iSeries 종료점 시스템으로 전송했습니다.

이 인증서를 *SIGNATUREVERIFICATION 저장소로 가져오려면 다음 단계를 수행하십시오.

1. DCM 탐색 프레임에서 **인증서 저장소 선택**을 클릭한 다음 ***SIGNATUREVERIFICATION**을 열 인증서 저장소로 선택하십시오.
2. 인증서 저장소 및 암호 페이지가 표시되면 인증서 저장소를 작성할 때 인증서 저장소에 지정한 암호를 제공한 다음 **계속**을 클릭하십시오.
3. 탐색 프레임의 화면을 정리한 후 **인증서 관리**를 선택하여 타스크 리스트를 표시하십시오.
4. 타스크 리스트에서 **인증서 가져오기**를 선택하십시오.
5. 인증서 유형으로 **CA(Certificate Authority)**를 선택한 다음 **계속**을 클릭하십시오.

   주: 개인 서명 확인 인증서를 가져오기 전에 로컬 CA 인증서를 가져와야 합니다. 그렇지 않으면 서명 확인 인증서의 가져오기 프로세스에서 오류가 발생합니다.
6. CA 인증서 파일의 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하십시오. 가져오기 프로세스가 성공했음을 확인하는 메시지나 프로세스가 실패한 경우 오류 정보를 제공하는 메시지가 표시됩니다.
7. **인증서 가져오기** 타스크를 다시 선택하십시오.
8. 가져올 인증서 유형으로 **서명 확인**을 선택하고 **계속**을 클릭하십시오.

9. 서명 확인 인증서 파일의 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하십시오. 가져오기 프로세스가 성공했음을 확인하는 메시지나 프로세스가 실패한 경우 오류 정보를 제공하는 메시지가 표시됩니다.

이제는 iSeries B에서 DCM을 사용하여 iSeries A에서 해당하는 서명 인증서로 작성한 오브젝트의 서명을 확인할 수 있습니다.

10단계: 서명 확인 타스크: 프로그램 오브젝트의 서명 확인

DCM을 사용하여 전송된 프로그램 오브젝트의 서명을 확인하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **인증서 저장소 선택**을 클릭한 다음 ***SIGNATUREVERIFICATION**을 열 인증서 저장소로 선택하십시오.
2. *SIGNATUREVERIFICATION 인증서 저장소에 대한 암호를 입력한 다음 **계속**을 클릭하십시오.
3. 탐색 프레임의 화면을 정리한 후 **서명 가능한 오브젝트 관리**를 선택하여 타스크 리스트를 표시하십시오.
4. 타스크 리스트에서 **오브젝트 서명 확인**을 선택하여 서명을 확인할 오브젝트의 위치를 지정하십시오.
5. 제공된 필드에서 오브젝트의 완전한 경로 및 파일명 또는 서명을 확인할 오브젝트의 디렉토리를 입력한 다음 **계속**을 클릭하십시오. 또는 디렉토리 위치를 입력하고 **찾아보기**를 클릭하여 디렉토리의 내용을 보면서 서명 확인할 오브젝트를 선택하십시오.

   주: 확인할 디렉토리의 일부를 나타내기 위해 특정 와일드카드 문자를 사용할 수도 있습니다. 이런 와일드카드 문자에는 *임의의 수의 문자*를 지정하는 별표(*****)와 *하나의 문자*를 지정하는 물음표(**?**)가 있습니다. 예를 들어, 특정 디렉토리의 모든 오브젝트에 서명하려면 `/mydirectory/*;`를 입력할 수 있고, 특정 라이브러리의 모든 프로그램에 서명하려면 `/QSYS.LIB/QGPL.LIB/*.PGM`을 입력할 수 있습니다. 이 와일드카드는 경로명의 마지막 부분에만 사용할 수 있습니다. 예를 들어, `/mydirectory*/filename`을 입력하면 오류 메세지가 표시됩니다. 라이브러리 리스트나 디렉토리 내용을 보기 위해 찾아보기 기능을 사용하려면 **찾아보기**를 클릭하기 전에 경로명의 일부로 와일드카드를 입력해야 합니다.
6. 선택한 오브젝트의 서명을 확인하기 위해 사용할 처리 옵션을 선택한 다음 **계속**을 클릭하십시오.

   주: 작업 결과를 기다리기로 선택하면 결과 파일이 직접 브라우저로 표시됩니다. 현재 작업의 결과가 결과 파일 끝에 추가됩니다. 따라서 파일에는 현재 작업의 결과 외에도 이전 작업의 결과가 포함될 수 있습니다. 파일의 날짜 필드를 사용하여 파일의 어떤 행이 현재 작업에 적용되었는지 확인할 수 있습니다. 날짜 필드의 형식은 YYYYMMDD입니다. 파일의 첫 번째 필드는 메시지 ID(오브젝트 처리 중에 오류가 발생한 경우)이거나 날짜 필드(작업이 처리된 날짜를 가리킴)일 수 있습니다.
7. 서명 확인 작업의 결과를 저장하는 데 사용할 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하십시오. 또는 디렉토리 위치를 입력하고 **찾아보기**를 클릭하여 디렉토리의 내용을 보면서 작업 결과를 저장할 파일을 선택하십시오. 오브젝트 서명을 확인하기 위해 작업이 제출되었음을 나타내는 메시지가 표시됩니다. 작업 결과를 보려면 작업 기록부의 **QOBJSGNBAT** 작업을 참조하십시오.

시나리오: 오브젝트에 서명하고 오브젝트 서명을 확인하기 위해 API 사용

상황

MyCo, Inc.는 고객을 위해 어플리케이션을 개발하는 iSeries 협력업체입니다. 당신은 회사의 소프트웨어 개발자로서 이 어플리케이션을 패키지하여 고객에게 분배하는 책임이 있습니다. 현재 당신은 프로그램을 사용하여 어플리케이션을 패키지합니다. 고객은 CD(CD-ROM)를 주문하거나 회사 웹 사이트를 방문하여 어플리케이션을 다운로드할 수 있습니다.

당신은 산업 정보, 특히 보안에 대한 정보에 항상 귀를 기울입니다. 따라서 고객이 받거나 다운로드하는 프로그램 소스와 내용에 대한 고객의 관심에 대해 잘 알고 있습니다. 때때로 고객은 자신들이 신뢰할 수 있는 소스이지만 올바른 제품 소스가 아닌 곳에서 제품을 받거나 다운로드하고 있다고 생각하는 경우가 있습니다. 이러한 혼란으로 인해 가끔씩 원하는 제품이 아닌 다른 제품을 설치할 수 있습니다. 때로는 설치한 제품이 불법 프로그램으로 밝혀지거나 변경된 것일 수도 있고 시스템에 손상을 주기도 합니다.

iSeries 고객에게 이러한 유형의 문제가 흔하게 발생하는 것은 아니지만, 귀사로부터 구입한 어플리케이션이 정말 귀사의 어플리케이션인지 고객에게 보증하려고 합니다. 또한 고객에게 이 어플리케이션의 무결성을 검사하는 방법을 제공하여 고객이 어플리케이션을 설치하기 전에 어플리케이션의 변경 여부를 판별할 수 있도록 조치하려고 합니다.

조사 내용을 바탕으로, 보안 목표를 달성하기 위해 OS/400 오브젝트 서명 기능을 사용하기로 결정했습니다. 어플리케이션에 디지털 서명하면 고객들은 귀사가 자신이 받거나 다운로드한 어플리케이션의 합법적인 소스인지 확인할 수 있습니다. 현재 프로그래밍 방식으로 어플리케이션을 패키지하기 때문에 API를 사용하여 오브젝트 서명을 기존 패키지 프로세스에 쉽게 추가할 수 있다고 판단했습니다. 또한 고객이 귀사의 제품을 설치할 때 서명 확인 프로세스를 알 수 있도록 공용 인증서를 사용하여 오브젝트에 서명하기로 결정합니다.

어플리케이션 패키지의 일부로 오브젝트를 서명하기 위해 사용한 디지털 인증서의 사본을 포함시킵니다. 고객이 어플리케이션 패키지를 구입한 경우 인증서의 공용 키를 사용하여 어플리케이션의 서명을 확인할 수 있습니다. 이 프로세스를 통해 고객은 어플리케이션 오브젝트의 내용이 서명한 이후로 변경되지 않았음을 확인할 수 있으며 어플리케이션의 소스를 식별하고 확인할 수 있습니다.

이 예제는 다른 사용자가 사용할 수 있도록 개발하고 패키지하는 어플리케이션의 오브젝트를 프로그래밍 방식으로 서명하는 단계에 대한 유용한 개요로 사용할 수 있습니다.

시나리오 장점

이 시나리오에는 다음과 같은 장점이 있습니다.

- 패키지를 작성하고 오브젝트를 프로그래밍 방식으로 서명하는 데 API를 사용하면 이 보안을 구현하는 데 걸리는 시간을 줄일 수 있습니다.
- 오브젝트를 패키지하는 경우 오브젝트에 서명하는 데 API를 사용하면 오브젝트에 서명하기 위해 수행해야 할 단계의 수를 줄일 수 있습니다. 그 이유는 서명 프로세스가 패키지 프로세스의 일부이기 때문입니다.

- 오브젝트의 패키지에 서명하면 서명한 후 오브젝트가 변경되었는지 여부를 더 쉽게 판별할 수 있습니다. 그러면 고객의 어플리케이션 문제를 조사하기 위해 향후 수행하게 될 문제 해결 조치의 일부를 줄일 수 있습니다.
- 잘 알려진 공용 CA(Certificate Authority)의 인증서를 사용하여 오브젝트에 서명하면 제품 설치 프로그램에서 나감 프로그램의 일부로 Add Verifier API를 사용할 수 있습니다. 이 API를 사용하면 어플리케이션에 서명하기 위해 사용한 공용 인증서를 고객의 시스템에 자동으로 추가할 수 있습니다. 그러면 고객도 서명 확인을 확실히 알 수 있습니다.

목표

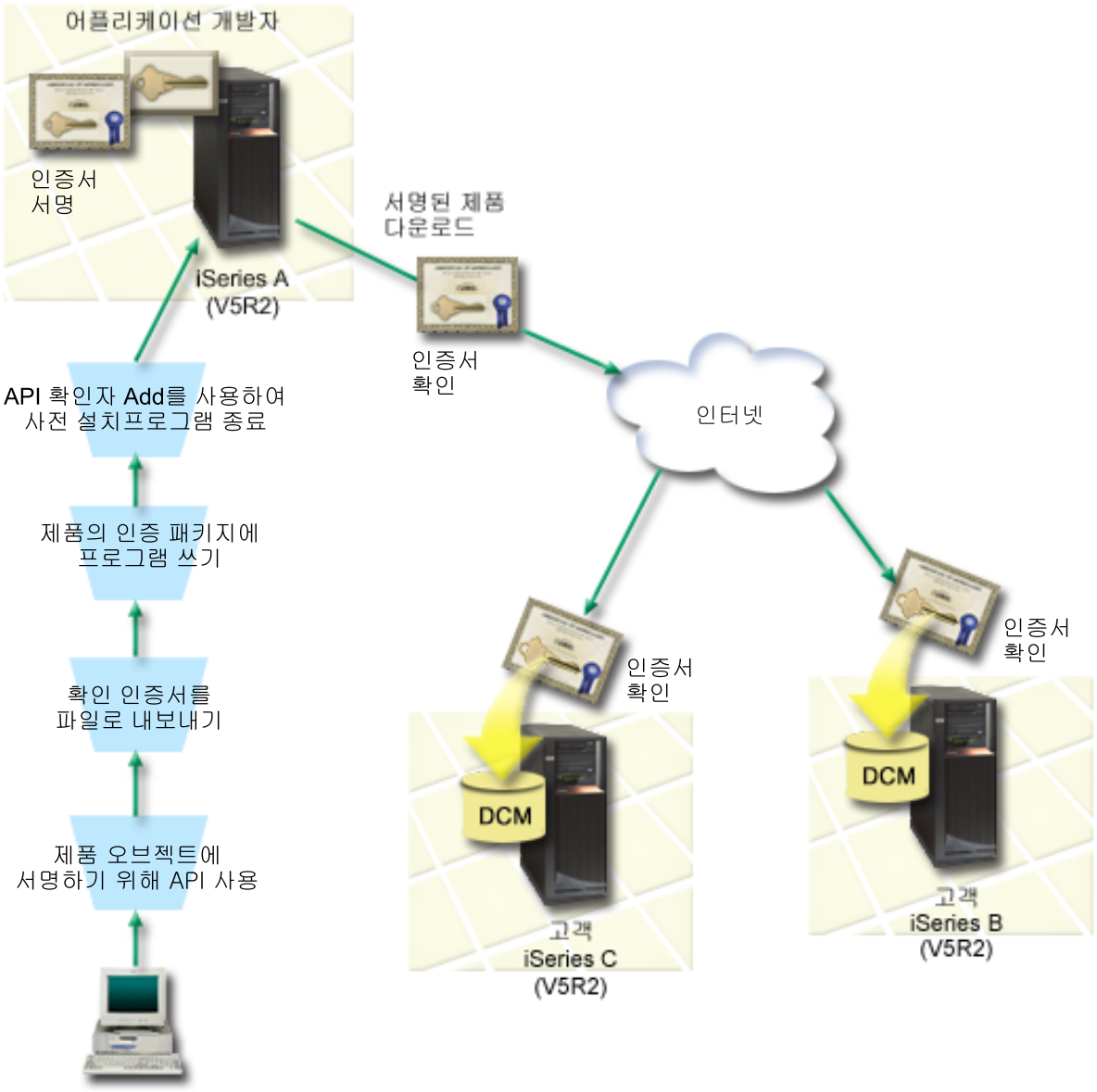
이 시나리오에서 MyCo, Inc.는 고객에게 패키지하거나 분배하는 어플리케이션에 프로그래밍 방식으로 서명하려고 합니다. MyCo, Inc.의 어플리케이션 프로덕션 개발자로서, 당신은 현재 회사의 어플리케이션을 프로그래밍 방식으로 패키지하여 고객에게 분배하고 있습니다. 따라서 iSeries API를 사용하여 어플리케이션에 서명하고 제품 설치 중에 고객의 iSeries에서 프로그래밍 방식으로 서명을 확인할 수 있게 하려고 합니다.

이 시나리오의 목표는 다음과 같습니다.

- 회사 프로덕션 개발자는 Sign Object API를 기존의 프로그래밍 어플리케이션 패키지 프로세스의 일부로 사용하여 오브젝트에 서명할 수 있어야 합니다.
- 어플리케이션 제품 설치 프로세스 중에 고객에게 서명 확인 프로세스를 알 수 있게 하려면 회사 어플리케이션을 공용 인증서로 서명해야 합니다.
- 회사는 iSeries API를 사용하여 프로그래밍 방식으로 필수 서명 확인 인증서를 고객의 iSeries 서버 *SIGNATUREVERIFICATION 인증서 저장소에 추가할 수 있어야 합니다. 회사는 제품 설치 프로세스의 일부로 고객의 iSeries 서버에서 이 인증서 저장소를 프로그래밍 방식으로 작성할 수 있어야 합니다.
- 고객은 제품 설치 후 어플리케이션의 디지털 서명을 보다 쉽게 확인할 수 있어야 합니다. 고객은 서명 확인을 통해 어플리케이션에 서명한 후 어플리케이션이 변경되었는지 여부를 확인할 수 있어야 하며 서명한 어플리케이션의 소스와 신뢰성을 확인할 수 있어야 합니다.

세부사항

다음 그림은 이 시나리오를 구현하기 위한 오브젝트 서명 및 서명 확인 프로세스를 보여줍니다.



이 그림은 이 시나리오와 관련된 다음 사항을 설명합니다.

중앙 시스템(iSeries A)

- iSeries A는 OS/400 버전 5 릴리스 2(V5R2)를 실행합니다.
- iSeries A는 어플리케이션 개발자의 제품 패키지 프로그램을 실행합니다.
- iSeries A에는 iSeries용 Cryptographic Access Provider 128비트(5722-AC3)가 설치되어 있습니다.
- iSeries A에는 디지털 인증 관리자(OS/400 옵션 34)와 IBM HTTP 서버(5722-DG1)가 설치 및 구성되어 있습니다.
- iSeries A는 회사 어플리케이션 제품의 1차 오브젝트 서명 시스템입니다. 다음 타스크를 수행하여 iSeries A에서 고객 분배를 위한 제품 오브젝트 서명을 완료할 수 있습니다.

1. API를 사용하여 회사 어플리케이션 제품에 서명하십시오.
2. DCM을 사용하여 고객이 서명된 오브젝트를 확인할 수 있게 파일에 서명 확인 인증서를 내보내십시오.
3. 서명된 어플리케이션 제품에 확인 인증서를 추가할 수 있게 프로그램을 작성하십시오.
4. ADD Verifier API를 사용하는 제품을 위해 사전 설치 나감 프로그램을 작성하십시오. 이 API를 사용하면 제품 설치 프로세스에서 확인 인증서를 고객 iSeries 서버(iSeries B 및 C)의 *SIGNATUREVERIFICATION 인증서 저장소에 프로그래밍 방식으로 추가할 수 있습니다.

고객 iSeries 서버 B 및 C

- iSeries B는 OS/400 버전 5 릴리스 2(V5R2)를 실행합니다.
- iSeries C는 OS/400 버전 5 릴리스 2(V5R2)를 실행합니다.
- iSeries B와 C에는 디지털 인증 관리자(옵션 34)와 IBM HTTP 서버(5722-DG1)가 설치 및 구성되어 있습니다.
- iSeries B와 C에서는 iSeries A를 소유한 어플리케이션 개발 회사의 웹 사이트에서 어플리케이션을 구매하고 다운로드합니다.
- MyCo의 어플리케이션 설치 프로세스에서 이 고객의 iSeries 서버에 *SIGNATUREVERIFICATION 인증서 저장소를 작성할 때 iSeries B와 C에서는 MyCo의 서명 확인 인증서 사본을 가져옵니다.

전제조건 및 가정

이 시나리오는 다음 전제조건과 가정에 따라 달라집니다.

1. 모든 iSeries 서버가 DCM(Digital Certificate Manager)을 설치하고 사용하기 위한 요구사항을 충족시킵니다.

   주: DCM을 설치 및 사용하기 위한 전제조건을 충족시키는 것은 고객(이 시나리오에서는 iSeries B 및 C)의 선택적인 요구사항입니다. Add Verifier API는 제품 설치 프로세스의 일부로 *SIGNATUREVERIFICATION 인증서 저장소를 작성하지만 필요한 경우 디폴트 암호로 작성합니다. 권한 없는 액세스로부터 이 인증서 저장소를 보호하려면 고객은 DCM을 사용하여 디폴트 암호를 변경해야 합니다.
2. 이전에 다른 사용자가 iSeries 서버에 DCM을 구성하거나 사용하지 않았습니다.
3. 모든 iSeries 서버에 최상위 수준의 Cryptographic Access Provider 128비트 사용권 프로그램(5722-AC3)이 설치되어 있습니다.
4. 모든 시나리오 iSeries 서버에서 복원 중 오브젝트 서명 확인(QVFYOBJRST) 시스템 값의 디폴트 설정은 3이고, 이 설정이 변경되지 않았습니다. 디폴트 설정을 사용하면 사용자가 서명된 오브젝트를 복원할 때 서버에서 오브젝트 서명을 확인할 수 있습니다.
5. iSeries A의 시스템 관리자는 오브젝트를 서명할 수 있는 *ALLOBJ 사용자 프로파일 특수 권한이 있거나 오브젝트 서명 어플리케이션에 대한 사용자 프로파일의 권한이 부여되어야 합니다.
6. 시스템 관리자나 DCM으로 인증서 저장소를 작성하는 다른 사용자(프로그램 포함)는 *SECADM 및 *ALLOBJ 사용자 프로파일 특수 권한이 있어야 합니다.

7. 시스템 관리자나 다른 모든 iSeries 서버의 사용자가 오브젝트 서명을 확인하려면 *AUDIT 사용자 프로파일 특수 권한을 갖고 있어야 합니다.

타스크 단계

이 시나리오의 설명과 같이 오브젝트에 서명하려면 iSeries A에서 다음 타스크를 완료해야 합니다.

1. 필요한 모든 iSeries 제품을 설치 및 구성하려면 모든 전제조건 단계를 완료하십시오.
2. 잘 알려진 공용 CA(Certificate Authority)에서 오브젝트 서명 인증서를 가져오려면 DCM을 사용하여 인증서 요청을 작성하십시오.
3. DCM을 사용하여 오브젝트 서명 어플리케이션 정의를 작성하십시오.
4. DCM을 사용하여 서명된 오브젝트 서명 인증서를 가져오고 이를 오브젝트 서명 어플리케이션 정의에 할당하십시오.
5. DCM을 사용하여 오브젝트 서명 인증서를 서명 확인 인증서로 내보내서 고객이 이를 사용하여 어플리케이션 오브젝트의 서명을 확인할 수 있도록 하십시오.
6. 서명 확인 인증서 파일을 제품의 일부로 포함시키고 고객에게 분배하기 위해 어플리케이션을 패키지할 때 Sign Object API를 사용하여 어플리케이션에 서명할 수 있도록 어플리케이션 패키지 프로그램을 다시 쓰십시오.
7. 어플리케이션 패키지 프로세스의 일부로 Add Verifier API를 사용하는 사전 설치 나감 프로그램을 작성하십시오. 이 나감 프로그램을 사용하면 *SIGNATUREVERIFICATION 인증서 저장소를 작성하고 필수 서명 확인 인증서를 제품 설치 중에 고객의 iSeries 서버에 추가할 수 있습니다.
8. 고객이 DCM을 사용하여 iSeries 서버의 *SIGNATUREVERIFICATION 인증서 저장소에 대한 디폴트 암호를 재설정하게 하십시오.

구성 세부사항

OS/400 API를 사용하여 시나리오의 설명과 같이 오브젝트에 서명하려면 다음 타스크 단계를 완료하십시오.

1단계: 모든 전제조건 단계 완료

이 시나리오 구현을 위한 특정 구성 작업을 수행하려면 필요한 모든 iSeries 제품을 설치 및 구성하는 전제조건 타스크를 모두 완료해야 합니다.

2단계: DCM을 사용하여 잘 알려진 공용 CA에서 인증서 확보

이 시나리오에서는 이전에 DCM(Digital Certificate Manager)을 사용하여 인증서를 작성 및 관리하지 않았다고 가정합니다. 따라서 오브젝트 서명 인증서를 작성하기 위한 프로세스의 일부로 *OBJECTSIGNING 인증서 저장소를 작성해야 합니다. 이 인증서 저장소가 작성되면 오브젝트 서명 인증서를 작성 및 관리하는 데 필요한 타스크가 제공됩니다. 잘 알려진 공용 CA(Certificate Authority)에서 인증서를 가져오려면 DCM을 사용하여 식별 정보와 인증서의 공용-개인 키 쌍을 작성하고 이 정보를 CA에 제출하여 인증서를 가져와야 합니다.

오브젝트 서명 인증서를 가져오기 위해 잘 알려진 공용 CA에 제공해야 할 인증서 요청 정보를 작성하려면 다음 단계를 완료하십시오.

1. DCM을 시작하십시오.
2. DCM의 탐색 프레임에서 **신규 인증서 저장소 작성**을 선택하여 안내된 타스크를 시작하고 일련의 양식을 완료하십시오. 이 양식은 오브젝트에 서명하기 위해 사용할 수 있는 인증서 저장소와 인증서를 작성하는 프로세스를 안내합니다.

   주: 안내된 이 타스크에서 특정 양식을 완료하는 방법에 대해 의문이 있으면 페이지 위쪽에 있는 물음표 (**?**)를 선택하여 온라인 도움말에 액세스하십시오.
3. 작성할 인증서 저장소로 ***OBJECTSIGNING**을 선택한 다음 **계속**을 클릭하십시오.
4. **예**를 선택하여 *OBJECTSIGNING 인증서 저장소를 작성하는 프로세스의 일부로 인증서를 작성한 다음 **계속**을 클릭하십시오.
5. 새로운 인증서의 서명자로 **VeriSign 또는 다른 인터넷 CA(Certificate Authority)**를 선택한 다음 **계속**을 클릭하여 새로운 인증서의 식별 정보를 제공할 수 있는 양식을 표시하십시오.
6. 양식을 완료하고 **계속**을 클릭하여 확인 페이지를 표시하십시오. 이 확인 페이지에서는 인증서를 발행할 공용 CA(Certificate Authority)에 제공해야 할 인증서 요청 데이터를 표시합니다. 인증서 서명 요청(CSR) 데이터는 공용 키와 새 인증서에 대해 지정한 기타 정보로 구성됩니다.
7. 인증서를 요청하기 위해 공용 CA에서 요구하는 인증서 어플리케이션 양식이나 별도의 파일에 CSR 데이터를 복사하여 붙여 넣으십시오. Begin 및 End New Certificate Request 행을 포함하여 모든 CSR 데이터를 사용해야 합니다. 이 페이지에서 나가면 데이터가 손실되며 회복할 수 없습니다.
8. 인증서를 발행 및 서명하기로 선택한 CA로 어플리케이션 양식이나 파일을 전송하십시오.
9. 시나리오의 다음 타스크 단계로 계속 진행하기 전에 CA에서 서명 및 완료된 인증서를 리턴하기를 기다리십시오.

3단계: 오브젝트 서명 어플리케이션 정의 작성

인증서 요청을 잘 알려진 공용 CA에 전송했습니다. 이제 DCM을 사용하여 오브젝트를 서명하기 위해 사용할 수 있는 오브젝트 서명 어플리케이션을 정의할 수 있습니다. 어플리케이션 정의에서 반드시 실제 어플리케이션을 참조할 필요는 없습니다. 그러나 작성한 어플리케이션 정의에서 서명하려고 하는 오브젝트의 유형이나 그룹을 설명해야 합니다. 어플리케이션 ID를 인증서와 연관시켜 서명 프로세스를 가능하게 하려면 정의가 필요합니다.

DCM을 사용하여 오브젝트 서명 어플리케이션 정의를 작성하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **인증서 저장소 선택**을 클릭한 다음 ***OBJECTSIGNING**을 열 인증서 저장소로 선택하십시오.
2. 인증서 저장소 및 암호 페이지가 표시되면 인증서 저장소를 작성할 때 인증서 저장소에 지정한 암호를 제공한 다음 **계속**을 클릭하십시오.
3. 탐색 프레임에서 **어플리케이션 관리**를 선택하여 타스크 리스트를 표시하십시오.
4. 타스크 리스트에서 **어플리케이션 추가**를 선택하여 어플리케이션을 정의하기 위한 양식을 표시하십시오.

5. 양식을 완료하고 **추가**를 클릭하십시오.

CA로부터 서명된 인증서를 받으면 자신이 작성한 어플리케이션에 인증서를 할당할 수 있습니다.

4단계: 서명된 공용 인증서를 가져와서 이를 오브젝트 서명 어플리케이션에 할당

오브젝트 서명을 사용하기 위해 인증서를 가져와서 해당 인증서를 어플리케이션에 할당하려면 다음 단계를 수행하십시오.

1. DCM을 시작하십시오.
2. 탐색 프레임에서 **인증서 저장소 선택**을 클릭한 다음 ***OBJECTSIGNING**을 열 인증서 저장소로 선택하십시오.
3. 인증서 저장소 및 암호 페이지가 표시되면 인증서 저장소를 작성할 때 인증서 저장소에 지정한 암호를 제공한 다음 **계속**을 클릭하십시오.
4. 탐색 프레임의 화면을 정리한 후 **인증서 관리**를 선택하여 타스크 리스트를 표시하십시오.
5. 타스크 리스트에서 **인증서 가져오기**를 선택하여 서명된 인증서를 인증서 저장소로 가져오는 프로세스를 시작하십시오.

   주: 안내된 이 타스크에서 특정 양식을 완료하는 방법에 대해 의문이 있으면 페이지 위쪽에 있는 물음표 (**?**)를 선택하여 온라인 도움말에 액세스하십시오.
6. **인증서 관리** 타스크 리스트에서 **인증서 할당**을 선택하여 현재 인증서 저장소의 인증서 리스트를 표시하십시오.
7. 리스트에서 인증서를 선택하고 **어플리케이션에 할당**을 클릭하여 현재 인증서 저장소의 어플리케이션 정의 리스트를 표시하십시오.
8. 리스트에서 어플리케이션을 선택하고 **계속**을 클릭하십시오. 할당 선택에 대한 확인 메시지가 표시되거나 문제가 발생한 경우 오류 메시지가 표시됩니다.

이 타스크를 완료하면 이제 OS/400 API를 사용하여 어플리케이션 및 다른 오브젝트에 서명할 수 있습니다. 그러나 사용자나 다른 사용자가 서명을 확인할 수 있게 하려면 필요한 인증서를 파일로 내보내고 서명한 어플리케이션을 설치한 iSeries 서버에 이 인증서를 전송해야 합니다. 고객 iSeries 서버에서는 인증서를 사용하여 자신이 설치한 어플리케이션의 서명을 확인할 수 있습니다. Add Verifier API를 어플리케이션 설치 프로그램의 일부로 사용하여 고객을 위해 필요한 서명 확인 구성을 수행할 수 있습니다. 예를 들어, 고객의 iSeries 서버를 구성하기 위해 Add Verifier API를 호출하는 사전 설치 나감 프로그램을 작성할 수 있습니다.

5단계: 다른 iSeries 서버에서 서명을 확인할 수 있도록 인증서 내보내기

오브젝트 서명에서는 사용자나 다른 사용자가 서명의 신뢰성을 확인하고, 해당 서명을 사용하여 서명한 오브젝트가 변경되었는지 여부를 판별하는 데 사용할 수 있는 방법이 있어야 합니다. 오브젝트에 서명한 동일한 시스템에서 오브젝트 서명을 확인하려면 DCM을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 작성해야 합니다. 이 인증서 저장소에는 오브젝트 서명 인증서의 사본과 서명 인증서를 발행한 CA의 CA 인증서 사본이 모두 포함되어야 합니다.

다른 사용자가 서명을 확인할 수 있게 하려면 그들에게 오브젝트를 서명한 인증서의 사본을 제공해야 합니다. 로컬 CA(Certificate Authority)를 사용하여 인증서를 발행할 경우 로컬 CA 인증서의 사본도 제공해야 합니다.

DCM을 사용하여 오브젝트를 서명한 같은 시스템(이 시나리오에서는 iSeries A)에서 서명을 확인하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **신규 인증서 저장소 작성**을 선택한 다음 ***SIGNATUREVERIFICATION**을 새로 작성할 인증서 저장소로 선택하십시오.
2. **예**를 선택하여 기존의 오브젝트 서명 인증서를 신규 인증 저장소에 서명 확인 인증서로 복사하십시오.
3. 새로운 인증서 저장소의 암호를 지정하고 **계속**을 클릭하여 인증서 저장소를 작성하십시오. 오브젝트에 서명하기 위해 사용한 같은 시스템에서 이제는 DCM을 사용하여 오브젝트 서명을 확인할 수 있습니다.

DCM을 사용하여 오브젝트 서명 인증서의 사본을 서명 확인 인증서로 내보냄으로써 다른 사용자가 오브젝트 서명을 확인할 수 있게 하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **인증서 관리**를 선택한 다음 **인증서 내보내기** 타스크를 선택하십시오.
2. **오브젝트 서명**을 선택하여 내보낼 수 있는 오브젝트 서명 인증서 리스트를 표시하십시오.
3. 리스트에서 해당하는 오브젝트 서명 인증서를 선택한 다음 **내보내기**를 클릭하십시오.
4. 대상으로 **파일, 서명 확인 인증서로**를 선택한 다음 **계속**을 클릭하십시오.
5. 내보낸 서명 확인 인증서의 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하여 인증서를 내보내십시오.

이제는 이 파일을 제품에 대해 작성한 어플리케이션 설치 패키지에 추가할 수 있습니다. Add Verifier API를 설치 프로그램의 일부로 사용하여 이 인증서를 고객의 *SIGNATUREVERIFICATION 인증서 저장소에 추가할 수 있습니다. 이 인증서 저장소가 없으면 API에서도 이 인증서 저장소를 작성합니다. 그러면 제품 설치 프로그램에서 고객의 iSeries 서버에 어플리케이션 오브젝트를 복원할 때 제품 설치 프로그램이 이 오브젝트의 서명을 확인할 수 있습니다.

6단계: iSeries API를 사용하여 어플리케이션에 서명하기 위한 어플리케이션 패키지 프로그램 업데이트

이제 어플리케이션 패키지에 추가할 서명 확인 인증서 파일이 있습니다. 고객 분배를 위해 제품 라이브러리의 패키지를 작성할 때 기존의 어플리케이션을 작성 또는 편집하는 Sign Object API를 사용하여 제품 라이브러리에 서명할 수 있습니다.

어플리케이션 패키지 작성 프로그램의 일부로 Sign Object API를 사용하는 방법을 더 잘 이해하려면 다음 코드 예제를 검토하십시오. C로 작성된 이 예제 코드 조각은 완벽한 서명 및 패키지 프로그램이 아닌 Sign Object API를 호출하는 프로그램 부분의 예제입니다. 이 프로그램 예제를 사용하기로 선택한 경우 해당 예제를 사용자의 특정 요구사항에 맞게 변경하십시오. 보안상의 이유로 IBM에서는 제공된 디폴트 값을 사용하지 말고 프로그램 예제를 개별화시킬 것을 권장합니다.

주: IBM은 귀하에게 유사한 기능을 귀하의 특정 요구에 맞게 조정하여 생성할 수 있도록 모든 프로그래밍 코드 예제를 사용할 수 있는 비독점적인 저작권 사용권을 부여합니다. 모든 샘플 예제는 IBM에 의해 예시

목적으로만 제공됩니다. 이러한 예제는 모든 조건하에서 철저히 테스트된 것은 아닙니다. 따라서 IBM은 이들 프로그램의 신뢰성, 실용성 또는 기능에 대해 보증하거나 암시할 수 없습니다. 여기에 포함된 모든 프로그램은 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여 어떠한 종류의 보증 없이 "현상태대로" 제공됩니다.

Sign Object API를 어플리케이션 제품의 패키지 작성 프로그램 일부로 사용하려면 사용자의 요구에 맞게 이 코드 조각을 변경하십시오. 이 프로그램에 두 개의 매개변수를 전달해야 하는데, 서명할 라이브러리명과 오브젝트 서명 어플리케이션 ID의 이름입니다. 어플리케이션 ID는 대소문자를 구분하지만 라이브러리명은 구분하지 않습니다. 서명할 제품의 일부로 여러 라이브러리를 사용할 경우 작성한 프로그램에서 이 코드 조각을 여러 번 호출할 수 있습니다.

```
/* ---------------------------------------------------------------- */
/*                                                                  */
/* COPYRIGHT (C) IBM CORP. 2002                                     */
/*                                                                  */
/* Use Sign Object API to sign one or more libraries                */
/*                                                                  */
/* The API will digitally sign all objects in a specified library   */
/*                                                                  */
/*                                                                  */
/*                                                                  */
/* This material contains programming source code for your          */
/* consideration.  This example has not been thoroughly             */
/* tested under all conditions.  IBM, therefore, cannot             */
/* guarantee or imply reliability, serviceability, or function      */
/* of these programs.  All programs contained herein are            */
/* provided to you "AS IS".  THE IMPLIED WARRANTIES OF              */
/* MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE         */
/* ARE EXPRESSLY DISCLAIMED.  IBM provides no program services for  */
/* these programs and files.                                        */
/*                                                                  */
/*                                                                  */
/*                                                                  */
/* The parameters are:                                              */
/*                                                                  */
/*  char *   name of the library to sign                            */
/*  char *   name of the application ID                             */
/*                                                                  */

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

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

    /* parameters:

       char * library to sign objects in,
       char * application identifier to sign with

     */

    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;    /* return exceptions for any errors */


    /* ------------------------------------ */
    /* construct path name given library name */
    /* ------------------------------------ */
    memset(libname, '\00', 11); /* initialize library name */
    for(lib_length = 0;
        ((*(argv[1] + lib_length) != ' ') &&
         (*(argv[1] + lib_length) != '\00'));
        lib_length++);
    memcpy(argv[1], libname, lib_length); /* fill in library name */

    /* build path name parm for API call */
    sprintf(path_name, "/QSYS.LIB/%s.LIB/*", libname);
    path_length = strlen(path_name);

    /* ---------------------------- */
    /* find length of application id */
    /* ---------------------------- */
    for(applid_length = 0;
        ((*(argv[2] + applid_length) != ' ') &&
         (*(argv[2] + applid_length) != '\00'));
        applid_length++);

    /* ------------------------------- */
    /* sign all objects in this library */
    /* ------------------------------- */
    QYDOSGNO  (path_name,             /* path name to object         */
               &path_length,          /* length of path name         */
               "OBJN0100",            /* format name                 */
               argv[2],               /* application identifier (ID) */
               &applid_length,        /* length of  application ID   */
               "1",                   /* replace duplicate signature */
               multi_objects,         /* how to handle multiple
                                            objects                */
               &multiobj_length,      /* length of multiple objects
                                         structure to use
                                      (0=no mult.object structure)*/
               &error_code);          /* error code                  */



  return 0;

 }
```

7단계: Add Verifier API를 사용하는 사전 설치 나감 프로그램 작성

이제 어플리케이션에 서명하기 위한 프로그래밍 방식의 프로세스가 있으므로, Add Verifier API를 설치 프로그램의 일부로 사용하여 분배를 위한 최종 제품을 만들 수 있습니다. 예를 들어, Add Verifier API를 사전

설치 나감 프로그램의 일부로 사용하여 서명된 어플리케이션 오브젝트를 복원하기 전에 인증서 저장소에 인증서를 추가시킬 수 있습니다. 그러면 어플리케이션 오브젝트를 고객의 iSeries 서버에 복원할 때 설치 프로그램에서 어플리케이션 오브젝트의 서명을 확인할 수 있습니다.

주: 보안상의 이유로 이 API는 사용자가 CA(Certificate Authority) 인증서를 *SIGNATUREVERIFICATION 인증서 저장소에 넣도록 허용하지 않습니다. CA 인증서를 인증서 저장소에 추가한 경우 시스템은 해당 CA를 신뢰할 수 있는 인증서의 소스로 간주합니다. 따라서 시스템은 이 CA에서 발행한 인증서를 신뢰할 수 있는 소스에서 나온 인증서로서 처리합니다. 그러므로 CA 인증서를 인증서 저장소에 넣기 위해 API를 사용하여 설치 나감 프로그램을 작성할 수 없습니다. CA 인증서를 인증서 저장소에 추가하여 시스템이 신뢰하는 CA를 수동으로 확실하게 제어하려면 디지털 인증 관리자를 사용해야 합니다. 디지털 인증 관리자를 사용하면 시스템이 관리자가 신뢰하지 않은 소스에서 인증서를 가져올 가능성을 방지할 수 있습니다.

누구도 사용자의 허락 없이 API를 사용하여 확인 인증서를 *SIGNATUREVERIFICATION 인증서 저장소에 추가하지 못하도록 하려면 시스템에서 이 API가 작동하지 않도록 설정하십시오. SST(System Service Tool)를 사용하여 보안 관련 시스템 값의 변경을 허용하지 않으면 이렇게 할 수 있습니다.

어플리케이션 설치 프로그램의 일부로 Add Verifier API를 사용하는 방법을 더 잘 이해하려면 다음 사전 설치 나감 프로그램 코드 예제를 검토하십시오. C로 작성된 이 예제 코드 조각은 완전한 사전 설치 나감 프로그램이 아닌 Add Verifier API를 호출하는 프로그램 부분의 예제입니다. 이 프로그램 예제를 사용하기로 선택한 경우 사용자의 특정 요구사항에 맞게 변경하십시오. 보안상의 이유로 IBM에서는 제공된 디폴트 값을 사용하지 말고 프로그램 예제를 개별화시킬 것을 권장합니다.

주: IBM은 귀하에게 유사한 기능을 귀하의 특정 요구에 맞게 조정하여 생성할 수 있도록 모든 프로그래밍 코드 예제를 사용할 수 있는 비독점적인 저작권 사용권을 부여합니다. 모든 샘플 예제는 IBM에 의해 예시 목적으로만 제공됩니다. 이러한 예제는 모든 조건하에서 철저히 테스트된 것은 아닙니다. 따라서 IBM은 이들 프로그램의 신뢰성, 실용성 또는 기능에 대해 보증하거나 암시할 수 없습니다. 여기에 포함된 모든 프로그램은 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여 어떠한 종류의 보증 없이 ″현상태대로″ 제공됩니다.

고객이 제품을 설치할 때 필요한 서명 확인 인증서를 고객의 iSeries 서버에 추가하기 위해 Add Verifier API를 사전 설치 나감 프로그램의 일부로 사용하려면 사용자의 요구사항에 맞게 이 코드 조각을 변경하십시오.

```
/* ------------------------------------------------------------ */
/*                                                              */
/* COPYRIGHT (C) IBM CORP. 2002                                 */
/*                                                              */
/* Use Add Verifier API to add a certificate in the specified   */
/* IFS file to the *SIGNATUREVERIFICATION certificate store.    */
/*                                                              */
/* The API will create the certificate store if it does not exist. */
/* If the certificate store is created it will be given a default */
/* password that should be changed using DCM as soon as possible. */
/* This warning needs to be given to the owners of the system that */
/* use this program.                                            */
/*                                                              */
/*                                                              */
/*                                                              */
```

```
/* This material contains programming source code for your          */
/* consideration.  This example has not been thoroughly             */
/* tested under all conditions.  IBM, therefore, cannot             */
/* guarantee or imply reliability, serviceability, or function      */
/* of these programs.  All programs contained herein are            */
/* provided to you "AS IS".  THE IMPLIED WARRANTIES OF              */
/* MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE         */
/* ARE EXPRESSLY DISCLAIMED.  IBM provides no program services for  */
/* these programs and files.                                        */
/*                                                                  */
/*                                                                  */
/*                                                                  */
/* The parameters are:                                              */
/*                                                                  */
/*  char *   path name to IFS file that holds the certificate       */
/*  char *   certificate label to give certificate                  */
/*                                                                  */
/*                                                                  */
/*                                                                  */
/* ---------------------------------------------------------------- */

#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];

    /* find length of path name */
    for(pathname_length = 0;
        ((*(pathname + pathname_length) != ' ') &&
         (*(pathname + pathname_length) != '\00'));
        pathname_length++);

    /* find length of certificate label */
    for(cert_label_length = 0;
        ((*(certlabel + cert_label_length) != ' ') &&
         (*(certlabel + cert_label_length) != '\00'));
        cert_label_length++);


    error_code.Bytes_Provided = 0;     /* return exceptions for any errors  */

    QydoAddVerifier (pathname,         /* path name to file with certificate*/
                     &pathname_length, /* length of path name               */
                     "OBJN0100",       /* format name                       */
                     certlabel,        /* certificate label                 */
                     &cert_label_length, /* length of certificate label     */
                     &error_code);     /* error code                        */

  return 0;
}
```

이 타스크를 완료하면 어플리케이션을 패키지하여 고객에게 분배할 수 있습니다. 타스크에서 어플리케이션을 설치하면 설치 프로세스의 일부로 서명된 어플리케이션 오브젝트가 확인됩니다. 나중에 고객은 DCM(Digital Certificate Manager)을 사용하여 어플리케이션 오브젝트의 서명을 확인할 수 있습니다. DCM을 사용하면 고객이 어플리케이션의 소스가 신뢰할 수 있는 것인지와 어플리케이션에 서명한 후 어플리케이션이 변경되었는지 여부를 판별할 수 있습니다.

주: 설치 프로그램에서 고객에 대한 디폴트 암호를 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 작성했습니다. 권한 없는 액세스를 방지하기 위해 고객에게 가능한 한 빨리 DCM을 사용하여 인증서 저장소의 암호를 다시 설정하도록 조언해야 합니다.

8단계: 고객이 *SIGNATUREVERIFICATION 인증서 저장소의 디폴트 암호를 재설정

Add Verifier API에서 제품 설치 프로세스의 일부로 고객의 iSeries 서버에 *SIGNATUREVERIFICATION 인증서 저장소를 작성했습니다. API에서 인증서 저장소를 작성한 경우 인증서 저장소에 대한 디폴트 암호를 작성합니다. 따라서 권한 없는 액세스로부터 이 인증서 저장소가 보호되도록 고객에게 DCM을 사용하여 이 암호를 재설정하도록 조언해야 합니다.

고객에게 이 단계를 완료하여 *SIGNATUREVERIFICATION 인증서 저장소 암호를 재설정하게 하십시오.

1. DCM을 시작하십시오.
2. 탐색 프레임에서 **인증서 저장소 선택**을 클릭한 다음 ***SIGNATUREVERIFICATION**을 열 인증서 저장소로 선택하십시오.
3. 인증서 저장소 및 암호 페이지가 표시되면 **암호 재설정**을 클릭하여 인증서 저장소 암호 재설정 페이지를 표시하십시오.

   주: 안내된 이 타스크에서 특정 양식을 완료하는 방법에 대해 의문이 있으면 페이지 위쪽에 있는 물음표(**?**)를 선택하여 온라인 도움말에 액세스하십시오.
4. 저장소에 대한 새 암호를 지정하고 암호를 다시 입력하여 확인한 다음, 인증서 저장소에 대한 암호 만기 정책을 선택하고 **계속**을 클릭하십시오.

시나리오: 오브젝트에 서명하기 위해 중앙 관리 사용

상황

MyCo, Inc.는 회사 내 여러 위치에 있는 복수 iSeries 서버에 분배하는 어플리케이션을 개발하는 회사입니다. 네트워크 관리자로서, 당신은 모든 회사 iSeries 서버에 이 어플리케이션을 설치 및 갱신할 책임이 있습니다. 당신은 현재 iSeries Navigator의 중앙 관리 기능을 사용하여 이 어플리케이션을 보다 쉽게 패키지하고 분배하며, 다른 관리 타스크를 수행합니다. 그러나 오브젝트에 대한 권한 없는 변경 때문에 이 어플리케이션으로 문제를 조사 및 수정하는 데 시간이 더 걸립니다. 따라서 오브젝트에 디지털로 서명하여 오브젝트의 무결성을 보호하려고 합니다.

OS/400오브젝트 서명 기능에 대한 연구를 통해, V5R2부터 중앙 관리를 사용하면 오브젝트를 패키지 작성하고 분배할 때 오브젝트에 서명할 수 있다는 것을 알았습니다. 중앙 관리를 사용하면 회사의 보안 목표를 효율

적으로 그리고 비교적 쉽게 달성할 수 있습니다. 또한 로컬 CA(Certificate Authority)를 작성하고 그 로컬 CA로 인증서를 발행해서 오브젝트에 서명하기로 결정했습니다. 오브젝트 서명 시 로컬 인증 기관에서 발행한 개인 인증서를 사용하면 잘 알려진 공용 CA의 인증서를 구매하지 않아도되기 때문에 보안 기술 사용 비용을 줄일 수 있습니다.

이 예제는 여러 회사의 iSeries 서버로 분배하는 어플리케이션에 대해 오브젝트 서명 구성 및 사용에 관련된 단계에 대한 유용한 개요로 사용할 수 있습니다.

시나리오 장점

이 시나리오에는 다음과 같은 장점이 있습니다.

- 패키지를 작성하고 오브젝트에 서명하기 위해 중앙 관리를 사용하면 서명된 오브젝트를 회사의 iSeries 서버에 분배하는 데 걸리는 시간을 줄일 수 있습니다.
- 패키지의 오브젝트에 서명하기 위해 중앙 관리를 사용하면 오브젝트에 서명하기 위해 수행해야 할 단계의 수를 줄일 수 있습니다. 서명 프로세스가 패키지 작성 프로세스의 일부이기 때문입니다.
- 오브젝트의 패키지에 서명하면 서명한 후 오브젝트가 변경되었는지 여부를 더 쉽게 판별할 수 있습니다. 그러면 어플리케이션 문제를 조사하기 위해 향후 수행하게 될 문제 해결 조치의 일부를 줄일 수 있습니다.
- 오브젝트 서명 시 로컬 CA(Certificate Authority)에서 발행한 인증서를 사용하면 오브젝트 서명을 구현하는 데 드는 비용을 줄일 수 있습니다.

목표

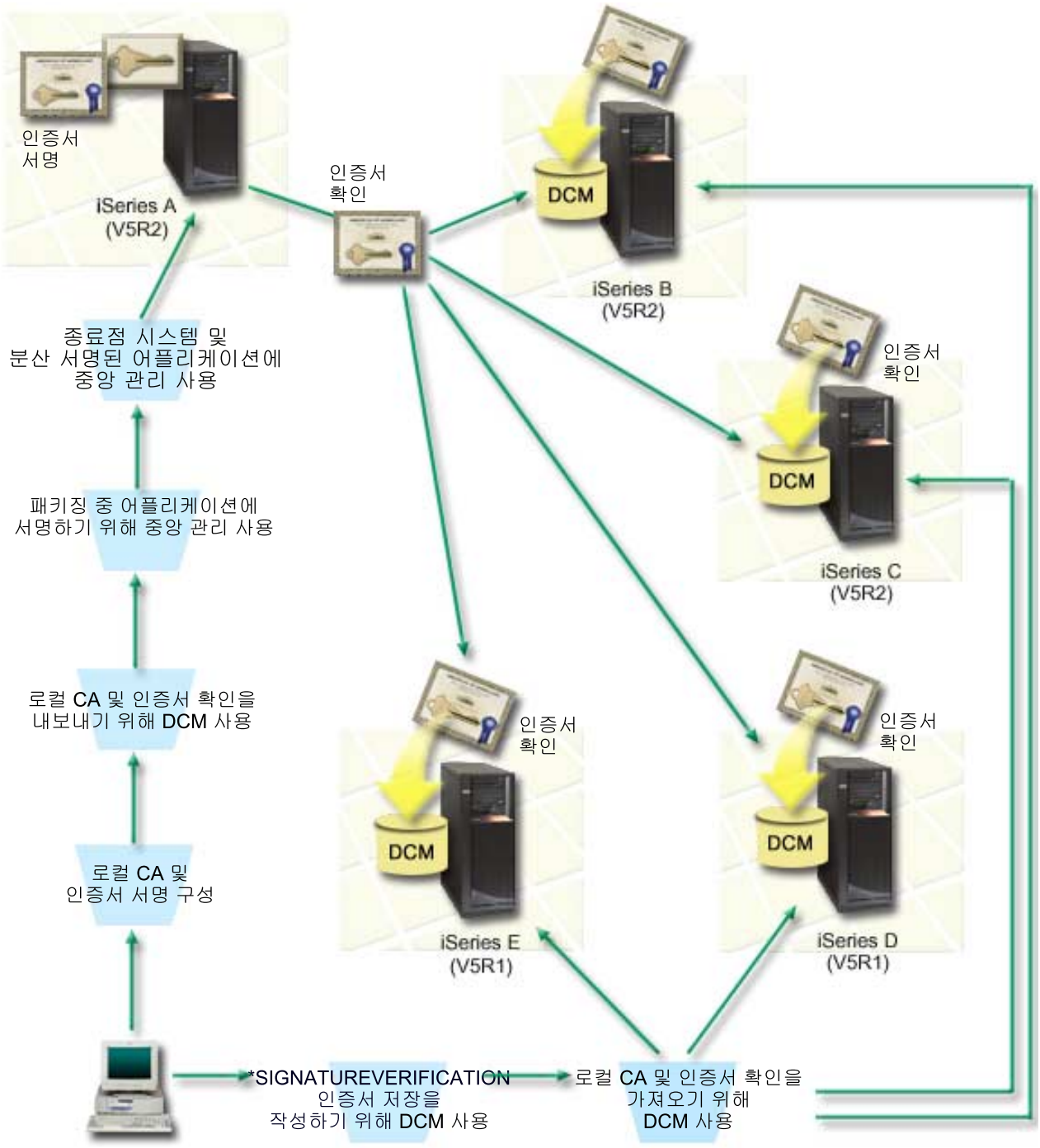
이 시나리오에서 MyCo, Inc.는 회사 내 여러 iSeries 서버에 분배하는 어플리케이션에 디지털 서명을 하려고 합니다. MyCo, Inc.의 네트워크 관리자로서, 당신은 이미 여러 iSeries 관리 타스크에 중앙 관리를 사용하고 있습니다. 따라서 다른 iSeries 서버에 분배하는 회사의 어플리케이션에 서명하기 위해 현재 사용하고 있는 중앙 관리를 확대하려고 합니다.

이 시나리오의 목표는 다음과 같습니다.

- 어플리케이션 서명 비용을 줄이기 위해 회사 어플리케이션을 로컬 CA에서 발행한 인증서로 서명해야 합니다.
- 시스템 관리자와 다른 지정한 사용자가 iSeries 서버의 디지털 서명을 쉽게 확인해서 회사의 서명한 오브젝트의 소스와 신뢰성을 확인할 수 있어야 합니다. 이를 달성하려면 모든 iSeries 서버에서는 회사의 서명 확인 인증서와 로컬 CA(Certificate Authority) 인증서의 사본이 모두 *SIGNATUREVERIFICATION 인증서 저장소에 있어야 합니다.
- 회사 어플리케이션의 서명을 확인하여 iSeries 관리자와 다른 사용자는 오브젝트를 서명한 후 오브젝트 내용이 변경되었는지 확인할 수 있습니다.
- 관리자는 중앙 관리를 사용하여 패키지를 작성 및 서명하고, 그런 다음 어플리케이션을 자신의 iSeries 서버에 분배할 수 있어야 합니다.

세부사항

다음 그림은 이 시나리오를 구현하기 위한 오브젝트 서명 및 서명 확인 프로세스를 보여줍니다.



이 그림은 이 시나리오와 관련된 다음 사항을 설명합니다.

중앙 시스템(iSeries A)

- iSeries A는 OS/400 버전 5 릴리스 2(V5R2)를 실행합니다.
- iSeries A는 회사 어플리케이션의 패키지 작성 및 분배를 포함하여 중앙 관리 기능이 실행되는 중앙 시스템 역할을 합니다.

- iSeries A에는 iSeries용 Cryptographic Access Provider 128비트(5722-AC3)가 설치되어 있습니다.
- iSeries A에는 디지털 인증 관리자(OS/400옵션 34) 및 IBM HTTP 서버(5722-DG1)가 설치 및 구성되어 있습니다.
- iSeries A는 로컬 CA(Certificate Authority) 역할을 하며 오브젝트 서명 인증서가 이 시스템에 상주합니다.
- iSeries A는 회사 어플리케이션의 1차 오브젝트 서명 시스템입니다. 다음 타스크를 수행하여 iSeries A에서 고객 분배를 위한 제품 오브젝트 서명을 완료할 수 있습니다.
  1. DCM을 사용하여 로컬 CA를 작성하고, 로컬 CA를 사용하여 오브젝트 서명 인증서를 작성하십시오.
  2. DCM을 사용하여 로컬 CA 인증서와 서명 확인 인증서의 사본을 파일로 내보냄으로써 종료점 시스템(iSeries B, C, D 및 E)에서 서명된 오브젝트를 확인할 수 있도록 하십시오.
  3. 중앙 관리를 사용하여 어플리케이션 오브젝트에 서명하고 확인 인증서 파일로 어플리케이션 오브젝트를 패키지하십시오.
  4. 중앙 관리를 사용하여 서명된 어플리케이션과 인증서 파일을 종료점 시스템으로 분배하십시오.

종료점 시스템(iSeries 서버 B, C, D 및 E)

- iSeries B와 C는 OS/400 버전 5 릴리스 2(V5R2)를 실행합니다.
- iSeries D와 E는 OS/400 버전 5 릴리스 1(V5R1)을 실행합니다.
- iSeries B, C, D, E에는 디지털 인증 관리자(옵션 34)와 IBM HTTP 서버(5722-DG1)가 설치 및 구성되어 있습니다.
- 시스템에서 서명된 어플리케이션을 수신할 경우 iSeries B, C, D, E는 회사의 서명 확인 인증서 사본과 중앙 시스템(iSeries A)의 로컬 CA 사본을 모두 수신합니다.
- DCM을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 작성하고 로컬 CA와 확인 인증서를 이 인증서 저장소로 가져옵니다.

전제조건 및 가정

이 시나리오는 다음 전제조건과 가정에 따라 달라집니다.

1. 모든 iSeries 서버가 DCM(Digital Certificate Manager)을 설치하고 사용하기 위한 요구사항을 충족시킵니다.
2. 이전에 다른 사용자가 iSeries 서버에 DCM을 구성하거나 사용하지 않았습니다.
3. iSeries A가 iSeries Navigator 및 중앙 관리의 설치 및 사용을 위한 요구사항을 충족시킵니다.
4. 모든 iSeries 종료점 시스템에서 중앙 관리 서버를 실행해야 합니다.
5. 모든 iSeries 서버에 최상위 수준의 Cryptographic Access Provider 128비트 사용권 프로그램(5722-AC3)이 설치되어 있습니다.
6. 모든 시나리오 iSeries 서버에서 복원 중 오브젝트 서명 확인(QVFYOBJRST) 시스템 값의 디폴트 설정은 3이고, 이 설정이 변경되지 않았습니다. 디폴트 설정을 사용하면 사용자가 서명된 오브젝트를 복원할 때 서버에서 오브젝트 서명을 확인할 수 있습니다.

7. iSeries A의 네트워크 관리자는 오브젝트를 서명할 수 있는 *ALLOBJ 사용자 프로파일 특수 권한이 있거나 오브젝트 서명 어플리케이션에 대한 사용자 프로파일의 권한이 부여되어야 합니다.
8. DCM에서 인증서 저장소를 작성하는 네트워크 관리자나 다른 사용자는 *SECADM 및 *ALLOBJ 사용자 프로파일특수 권한이 있어야 합니다.
9. 시스템 관리자나 다른 모든 iSeries 서버의 사용자가 오브젝트 서명을 확인하려면 *AUDIT 사용자 프로파일 특수 권한이 있어야 합니다.

타스크 단계

이 시나리오를 구현하려면 완료해야 할 두 가지 타스크 세트가 있습니다. 첫 번째 타스크 세트를 사용하면 iSeries A에서 중앙 관리를 사용하여 어플리케이션에 서명 및 분배할 수 있습니다. 다른 타스크 세트를 사용하면 시스템 관리자와 다른 사용자가 다른 모든 iSeries 서버에서 이 어플리케이션의 서명을 확인할 수 있습니다.

오브젝트 서명 타스크 단계

이 시나리오의 설명과 같이 오브젝트에 서명하려면 iSeries A에서 다음 타스크를 완료해야 합니다.

1. 필요한 모든 iSeries 제품을 설치 및 구성하려면 모든 전제조건 단계를 완료하십시오.
2. DCM(Digital Certificate Manager)을 사용하여 개인 오브젝트 서명 인증서를 발행하기 위한 로컬 CA(Certificate Authority)를 작성하십시오.
3. DCM을 사용하여 어플리케이션 정의를 작성하십시오.
4. DCM을 사용하여 오브젝트 서명 어플리케이션 정의에 인증서를 할당하십시오.
5. DCM을 사용하여 오브젝트 서명을 확인하기 위해 다른 시스템에서 사용해야 할 인증서를 내보내십시오. 로컬 CA 인증서의 사본과 오브젝트 서명 인증서의 사본을 서명 확인 인증서로 파일에 내보내야 합니다.
6. 서명을 확인할 각 iSeries 종료점 시스템에 인증서 파일을 전송하십시오.
7. 어플리케이션 오브젝트에 서명하려면 중앙 관리를 사용하십시오.

서명 확인 타스크 단계

중앙 관리를 사용하여 서명된 어플리케이션 오브젝트를 iSeries 종료점 시스템에 전송하기 전에 모든 iSeries 종료점 시스템에서 이 서명 확인 구성 타스크를 완료해야 합니다. 서명된 오브젝트를 종료점 시스템에 복원할 때 서명을 확인하려면 서명 확인 구성을 완료해야 합니다.

모든 iSeries 종료점 시스템에서 시나리오에서 설명한 대로 오브젝트의 서명을 확인하려면 다음 타스크를 완료해야 합니다.

8. DCM(Digital Certificate Manager)을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 작성하십시오.
9. DCM을 사용하여 로컬 CA 인증서와 서명 확인 인증서를 가져오십시오.

구성 세부사항

중앙 관리를 구성하기 위한 다음 타스크 단계를 완료하여 시나리오에서 설명한 대로 오브젝트에 서명하십시오.

1단계: 모든 전제조건 단계 완료

이 시나리오 구현을 위한 특정 구성 타스크를 수행하려면 필요한 모든 iSeries 제품을 설치 및 구성하는 전제조건 타스크를 모두 완료해야 합니다.

2단계: 개인 오브젝트 서명 인증서를 발행할 로컬 CA 작성

DCM(Digital Certificate Manager)을 사용하여 로컬 CA(Certificate Authority)를 작성할 경우 프로세스에서 일련의 양식을 완료해야 합니다. 이 양식은 SSL(Secure Sockets Layer), 오브젝트 서명 및 서명 확인을 위한 디지털 인증서를 사용하기 위해 완료해야 하는 타스크와 CA를 작성하기 위한 프로세스를 안내해줍니다. 이 시나리오에서는 SSL용 인증서를 구성하지 않아도 되지만 시스템이 오브젝트를 서명할 수 있도록 구성하는 타스크에서 모든 양식을 완료해야 합니다.

DCM을 사용하여 로컬 CA를 작성한 후 작동시키려면 다음 단계를 수행하십시오.

1. DCM을 시작하십시오.
2. DCM의 탐색 프레임에서 **CA(Certificate Authority) 작성**을 선택하여 일련의 양식을 표시하십시오.

   주: 안내된 이 타스크에서 특정 양식을 완료하는 방법에 대해 의문이 있으면 페이지 위쪽에 있는 물음표 (**?**) 버튼을 선택하여 온라인 도움말에 액세스하십시오.
3. 안내된 이 타스크의 모든 양식을 완료하십시오. 이 타스크를 수행할 때 다음을 수행해야 합니다.
   a. 로컬 CA에 식별 정보를 제공하십시오.
   b. 브라우저에 로컬 CA 인증서를 설치하여 소프트웨어에서 로컬 CA를 인식하고 로컬 CA에서 발행하는 인증서를 확인할 수 있게 하십시오.
   c. 로컬 CA에 대한 정책 데이터를 지정하십시오.
   d. 새로운 로컬 CA를 사용하여 어플리케이션에서 SSL 연결에 사용할 수 있는 서버나 클라이언트 인증서를 발행하십시오.

      주: 이 시나리오에서는 이 인증서를 사용하지 않지만 인증서를 작성해야 로컬 CA를 사용하여 필요한 오브젝트 서명 인증서를 발행할 수 있습니다. 이 인증서를 작성하지 않고 타스크를 취소한 경우 오브젝트 서명 인증서와 오브젝트 서명 인증서가 저장되는 *OBJECTSIGNING 인증서 저장소를 각각 작성해야 합니다.
   e. SSL 연결을 위해 서버 인증서나 클라이언트 인증서를 사용할 수 있는 어플리케이션을 선택하십시오.

      주: 이 시나리오 대로 진행하려면 어플리케이션을 선택하지 말고 **계속**을 클릭하여 다음 양식을 표시하십시오.
   f. 새로운 로컬 CA를 사용하여 어플리케이션에서 오브젝트를 디지털 서명하기 위해 사용할 수 있는 오브젝트 서명 인증서를 발행하십시오. 이 서브타스크는 *OBJECTSIGNING 인증서 저장소를 작성합니다. 이것이 바로 오브젝트 서명 인증서를 관리하기 위해 사용하는 인증서 저장소입니다.
   g. 로컬 CA를 신뢰하는 어플리케이션을 선택하십시오.

주: 이 시나리오 대로 진행하려면 어플리케이션을 선택하지 말고 **계속**을 클릭하여 타스크를 완료하십시오.

이제 로컬 CA와 오브젝트 서명 인증서를 작성했습니다. 오브젝트를 서명하려면 인증서를 사용할 수 있도록 오브젝트 서명 어플리케이션을 정의해야 합니다.

3단계: 오브젝트 서명 어플리케이션 정의 작성

오브젝트 서명 인증서를 작성한 후에는 DCM(Digital Certificate Manager)을 사용하여 오브젝트에 서명하기 위해 사용할 수 있는 오브젝트 서명 어플리케이션을 정의해야 합니다. 어플리케이션 정의에서 반드시 실제 어플리케이션을 참조할 필요는 없습니다. 그러나 작성한 어플리케이션 정의에서 서명하려고 하는 오브젝트의 유형이나 그룹을 설명해야 합니다. 어플리케이션 ID를 인증서와 연관시켜 서명 프로세스를 가능하게 하려면 정의가 필요합니다.

DCM을 사용하여 오브젝트 서명 어플리케이션 정의를 작성하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **인증서 저장소 선택**을 클릭한 다음 ***OBJECTSIGNING**을 열 인증서 저장소로 선택하십시오.
2. 인증서 저장소 및 암호 페이지가 표시되면 인증서 저장소를 작성할 때 인증서 저장소에 지정한 암호를 제공한 다음 **계속**을 클릭하십시오.
3. 탐색 프레임에서 **어플리케이션 관리**를 선택하여 타스크 리스트를 표시하십시오.
4. 타스크 리스트에서 **어플리케이션 추가**를 선택하여 어플리케이션을 정의하기 위한 양식을 표시하십시오.
5. 양식을 완료하고 **추가**를 클릭하십시오.

이제 오브젝트 서명 인증서를 작성한 어플리케이션에 할당해야 합니다.

4단계: 오브젝트 서명 어플리케이션 정의에 인증서 할당

오브젝트 서명 어플리케이션에 인증서를 할당하려면 다음 단계를 수행하십시오.

1. DCM 탐색 프레임에서 **인증서 관리**를 선택하여 타스크 리스트를 표시하십시오.
2. 타스크 리스트에서 **인증서 할당**을 선택하여 현재 인증서 저장소의 인증서 리스트를 표시하십시오.
3. 리스트에서 인증서를 선택하고 **어플리케이션에 할당**을 클릭하여 현재 인증서 저장소의 어플리케이션 정의 리스트를 표시하십시오.
4. 리스트에서 하나 이상의 어플리케이션을 선택하고 **계속**을 클릭하십시오. 인증서 할당을 확인하는 메시지 페이지가 표시되거나 문제가 발생한 경우 오류 정보를 제공하는 메세지 페이지가 표시됩니다.

이 타스크를 완료하면 이제 오브젝트를 패키지하고 분배할 때 중앙 관리를 사용하여 오브젝트에 서명할 수 있습니다. 그러나 사용자나 다른 사용자가 서명을 확인할 수 있게 하려면 필요한 인증서를 파일로 내보내고 모든 iSeries 종료점 시스템에 이 인증서를 전송해야 합니다. 중앙 관리를 사용하여 서명된 어플리케이션 오브젝트를 iSeries 종료점 시스템에 전송하기 전에 각각의 iSeries 종료점 시스템에서 모든 서명 확인 구성 타스크를 완료해야 합니다. 서명된 오브젝트를 종료점 시스템에 복원할 때 서명을 확인하려면 서명 확인 구성을 완료해야 합니다.

5단계: 다른 iSeries 시스템에서 서명을 확인할 수 있도록 인증서 내보내기

내용의 무결성을 보호하기 위한 오브젝트 서명의 경우, 사용자나 기타 사용자에게는 서명의 신뢰성을 확인할 수 있는 수단이 있어야 합니다. 오브젝트에 서명한 동일한 시스템에서 오브젝트 서명을 확인하려면 DCM을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 작성해야 합니다. 이 인증서 저장소에는 오브젝트 서명 인증서의 사본과 서명 인증서를 발행한 CA의 CA 인증서 사본이 모두 포함되어야 합니다.

다른 사용자가 서명을 확인할 수 있게 하려면 그들에게 오브젝트를 서명한 인증서의 사본을 제공해야 합니다. 로컬 CA(Certificate Authority)를 사용하여 인증서를 발행할 경우 로컬 CA 인증서의 사본도 제공해야 합니다.

DCM을 사용하여 오브젝트를 서명한 같은 시스템(이 시나리오에서는 iSeries A)에서 서명을 확인하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **신규 인증서 저장소 작성**을 선택한 다음 ***SIGNATUREVERIFICATION**을 새로 작성할 인증서 저장소로 선택하십시오.
2. **예**를 선택하여 기존의 오브젝트 서명 인증서를 신규 인증 저장소에 서명 확인 인증서로 복사하십시오.
3. 새로운 인증서 저장소의 암호를 지정하고 **계속**을 클릭하여 인증서 저장소를 작성하십시오. 오브젝트에 서명하기 위해 사용한 같은 시스템에서 이제는 DCM을 사용하여 오브젝트 서명을 확인할 수 있습니다.

DCM을 사용하여 로컬 CA 인증서의 사본과 오브젝트 서명 인증서의 사본을 서명 확인 인증서로 내보냄으로써 다른 시스템에서 오브젝트 서명을 확인할 수 있게 하려면 다음 단계를 수행하십시오.

1. 탐색 프레임에서 **인증서 관리**를 선택한 다음 **인증서 내보내기** 타스크를 선택하십시오.
2. **CA(Certificate Authority)**를 선택하고 **계속**을 클릭하여 내보낼 수 있는 CA 인증서 리스트를 표시하십시오.
3. 리스트에서 이전에 작성한 로컬 CA 인증서를 선택하고 **내보내기**를 클릭하십시오.
4. 내보내기 대상으로 **파일**을 지정하고 **계속**을 클릭하십시오.
5. 내보낸 로컬 CA 인증서의 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하여 인증서를 내보내십시오.
6. **확인**을 클릭하여 내보내기 확인 페이지에서 나가십시오. 이제는 오브젝트 서명 인증서 사본을 내보낼 수 있습니다.
7. **인증서 내보내기** 타스크를 다시 선택하십시오.
8. **오브젝트 서명**을 선택하여 내보낼 수 있는 오브젝트 서명 인증서 리스트를 표시하십시오.
9. 리스트에서 해당하는 오브젝트 서명 인증서를 선택한 다음 **내보내기**를 클릭하십시오.
10. 대상으로 **파일, 서명 확인 인증서**를 선택한 다음 **계속**을 클릭하십시오.
11. 내보낸 서명 확인 인증서의 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하여 인증서를 내보내십시오.

이제 인증서로 작성한 서명을 확인할 iSeries 종료점 시스템으로 이 파일을 전송할 수 있습니다.

6단계: 인증서 파일을 iSeries 종료점 시스템으로 전송

서명한 오브젝트를 확인할 수 있도록 인증서 파일을 구성하려면 iSeries A에서 작성한 인증서 파일을 이 시나리오의 iSeries 종료점 시스템으로 전송해야 합니다. 여러 다른 방법을 사용하여 인증서 파일을 전송할 수 있습니다. 예를 들어, FTP(File Transfer Protocol)나 중앙 관리 패키지 분배를 사용하여 파일을 전송할 수 있습니다.

7단계: 중앙 관리를 사용하여 오브젝트에 서명

중앙 관리의 오브젝트 서명 프로세스는 소프트웨어 패키지 분배 프로세스의 일부입니다. 중앙 관리를 사용하여 서명된 어플리케이션 오브젝트를 iSeries 종료점 시스템에 전송하기 전에 각각의 iSeries 종료점 시스템에서 모든 서명 확인 구성 타스크를 완료해야 합니다. 서명된 오브젝트를 종료점 시스템에 복원할 때 서명을 확인하려면 서명 확인 구성을 완료해야 합니다.

이 시나리오에서 설명한 대로 iSeries 종료점 시스템으로 분배하는 어플리케이션에 서명하려면 다음 단계를 수행하십시오.

1. 중앙 관리를 사용하여 소프트웨어 제품을 패키지하여 분배하십시오.
2. **제품 정의** 마법사의 **ID** 패널이 표시되면 **고급**을 클릭하여 **고급 ID** 패널을 표시하십시오.
3. **디지털 서명** 필드에서, 이전에 작성한 오브젝트 서명 어플리케이션의 어플리케이션 ID를 입력한 다음 **확인**을 클릭하십시오.
4. 마법사를 완료하고 중앙 관리를 사용하여 소프트웨어 제품을 패키지 작성하고 분배하는 프로세스를 계속하십시오.

8단계: 서명 확인 타스크: iSeries 종료점 시스템에 *SIGNATUREVERIFICATION 인증서 저장소 작성

이 시나리오에서 iSeries 종료점 시스템의 오브젝트 서명을 확인하려면 각 시스템은 *SIGNATUREVERIFICATION 인증서 저장소에 해당하는 서명 확인 인증서의 사본이 있어야 합니다. 개인 인증서에서 오브젝트에 서명한 경우 이 인증서 저장소에 로컬 CA 인증서의 사본도 포함되어야 합니다.

*SIGNATUREVERIFICATION 인증서 저장소를 작성하려면 다음 단계를 수행하십시오.

1. DCM을 시작하십시오.
2. DCM(Digital Certificate Manager) 탐색 프레임에서 **신규 인증서 저장소 작성**을 선택한 다음 ***SIGNATUREVERIFICATION**을 새로 작성할 인증서 저장소로 선택하십시오.

   주: 안내된 이 타스크에서 특정 양식을 완료하는 방법에 대해 의문이 있으면 페이지 위쪽에 있는 물음표(**?**)를 선택하여 온라인 도움말에 액세스하십시오.
3. 새로운 인증서 저장소의 암호를 지정하고 **계속**을 클릭하여 인증서 저장소를 작성하십시오. 이제는 인증서를 저장소로 가져온 후 이 인증서를 사용하여 오브젝트 서명을 확인할 수 있습니다.

9단계: 서명 확인 타스크: 인증서 가져오기

오브젝트의 서명을 확인하려면 *SIGNATUREVERIFICATION 저장소에 서명 확인 인증서의 사본이 포함되어 있어야 합니다. 서명 인증서가 개인 인증서일 경우 이 인증서 저장소에 서명 인증서를 발행한 로컬 CA(Certificate Authority) 인증서의 사본도 있어야 합니다. 이 시나리오에서는 두 개의 인증서를 파일로 내보냈으며 해당 파일을 각각의 iSeries 종료점 시스템으로 전송했습니다.

이 인증서를 *SIGNATUREVERIFICATION 저장소로 가져오려면 다음 단계를 수행하십시오.

1. DCM 탐색 프레임에서 **인증서 저장소 선택**을 클릭한 다음 ***SIGNATUREVERIFICATION**을 열 인증서 저장소로 선택하십시오.
2. 인증서 저장소 및 암호 페이지가 표시되면 인증서 저장소를 작성할 때 인증서 저장소에 지정한 암호를 제공한 다음 **계속**을 클릭하십시오.
3. 탐색 프레임의 화면을 정리한 후 **인증서 관리**를 선택하여 타스크 리스트를 표시하십시오.
4. 타스크 리스트에서 **인증서 가져오기**를 선택하십시오.
5. 인증서 유형으로 **CA(Certificate Authority)**를 선택한 다음 **계속**을 클릭하십시오.

   주: 개인 서명 확인 인증서를 가져오기 전에 로컬 CA 인증서를 가져와야 합니다. 그렇지 않으면 서명 확인 인증서의 가져오기 프로세스에서 오류가 발생합니다.
6. CA 인증서 파일의 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하십시오. 가져오기 프로세스가 성공했음을 확인하는 메시지나 프로세스가 실패한 경우 오류 정보를 제공하는 메시지가 표시됩니다.
7. **인증서 가져오기** 타스크를 다시 선택하십시오.
8. 가져올 인증서 유형으로 **서명 확인**을 선택하고 **계속**을 클릭하십시오.
9. 서명 확인 인증서 파일의 완전한 경로 및 파일명을 지정한 다음 **계속**을 클릭하십시오. 가져오기 프로세스가 성공했음을 확인하는 메시지나 프로세스가 실패한 경우 오류 정보를 제공하는 메시지가 표시됩니다.

이제 iSeries 시스템은 사용자가 서명된 오브젝트를 복원할 때 해당하는 서명 인증서로 작성한 오브젝트의 서명을 확인할 수 있습니다.

오브젝트 서명 개념

iSeries 오브젝트 서명 및 서명 확인 기능을 사용하기 전에 이 개념 중 일부를 검토하는 것이 도움이 될 수 있습니다.

디지털 서명
디지털 서명과 디지털 서명에서 제공하는 보호에 대해 배웁니다.

서명 가능한 오브젝트
서명할 수 있는 iSeries 오브젝트와 명령(*CMD) 오브젝트 서명 옵션에 대해 배웁니다.

오브젝트 서명 처리
오브젝트 서명 처리의 작동 방법과 오브젝트 서명 처리를 위해 설정할 수 있는 매개변수에 대해 배웁니다.

서명 확인 처리
오브젝트 서명 확인의 작동 방법과 오브젝트 서명 확인을 위해 설정할 수 있는 매개변수에 대해 배웁니다.

디지털 서명

OS/400은 오브젝트의 디지털 ″서명″을 위해 디지털 인증서 사용에 대한 지원을 제공합니다. 오브젝트의 디지털 서명은 암호화 양식을 사용하여 작성되며 일반 문서의 개인 서명과 같습니다. 디지털 서명은 오브젝트의 원점에 대한 증거와 오브젝트의 무결성을 확인하는 수단을 제공합니다. 디지털 인증 소유자는 인증서의 개인 키를 사용하여 오브젝트에 ″서명″합니다. 오브젝트의 수신자는 인증서의 해당하는 공용 키를 사용하여 서명을 해독합니다. 서명 해독은 서명된 오브젝트의 무결성을 확인하고 소스인 송신자를 확인합니다.

오브젝트 서명은 오브젝트를 변경할 수 있는 사람을 제어하기 위한 일반적인 iSeries 서버 툴을 증대시킵니다. 일반적인 제어에서는 인터넷이나 다른 신뢰할 수 없는 네트워크를 통해 오브젝트를 전환하는 중에 오브젝트에 대한 권한 없는 변경을 막을 수 없습니다. 오브젝트에 서명한 후 오브젝트 내용이 변경되었는지 확인할 수 있기 때문에 이런 경우에 획득한 오브젝트를 신뢰할 수 있는지 여부를 더 쉽게 확인할 수 있습니다.

디지털 서명은 오브젝트 데이터의 암호화된 수리적인 요약입니다. 오브젝트와 오브젝트의 내용은 암호화되지 않고 디지털 서명에 의해 비공개되지만 요약 자체는 암호화되어 권한 없는 변경을 방지할 수 있습니다. 오브젝트가 전환 중에 변경되지 않았는지 확인하거나 오브젝트의 출처가 허용된 적합한 소스인지 확인하려면 서명 인증서의 공용 키 를 사용하여 원래 디지털 서명을 확인할 수 있습니다. 서명이 더 이상 일치하지 않으면 데이터가 변경되었을 수 있습니다. 이런 경우 수신자는 오브젝트 사용을 피하고 대신 서명자에게 문의하여 서명된 오브젝트의 다른 사본을 가져올 수 있습니다.

사용자는 오브젝트 서명을 위해 인증서를 사용할 수 있는 해당 권한이 필요하지만 오브젝트의 서명은 해당 시스템의 특정 사용자가 아닌 오브젝트에 서명한 시스템을 나타냅니다.

디지털 서명을 사용하는 것이 보안 요구사항과 정책에 적합하다고 결정한 경우 공용 인증서를 사용할지 여부 및 로컬 인증서 발행을 평가해야 합니다. 공개적으로 사용자에게 오브젝트를 분배할 경우 잘 알려진 공용 CA(Certificate Authority)의 인증서를 사용하여 오브젝트에 서명하는 것을 고려해야 합니다. 공용 인증서를 사용하면 다른 사용자에게 분배한 오브젝트의 서명을 다른 사용자가 쉽게 확인할 수 있습니다. 그러나 조직 내에서만 오브젝트를 분배할 경우 디지털 인증 관리자(DCM)를 사용하여 오브젝트 서명을 위한 인증서를 발행하도록 로컬 CA를 작동시킬 수 있습니다. 로컬 CA의 개인 인증서를 사용하여 오브젝트에 서명하는 것이 잘 알려진 공용 CA의 인증서를 구매하는 것보다 비용이 덜 듭니다.

디지털 서명의 유형

V5R2부터 명령(*CMD) 오브젝트에 서명할 수 있습니다. *CMD 오브젝트에 대한 두 가지 유형의 서명, 즉 핵심 오브젝트 서명과 전체 오브젝트 서명 중 하나를 선택할 수도 있습니다.

- **전체 오브젝트 서명**
  이 유형의 서명은 오브젝트의 비필수 바이트에만 적용됩니다.

- **핵심 오브젝트 서명**
  이 유형의 서명은 *CMD 오브젝트의 핵심 바이트에 적용됩니다. 그러나 자주 변경되는 바이트에는 적용되지 않습니다. 이 유형의 서명을 사용하면 서명을 무효화시키지 않고 명령을 변경할 수 있습니다. 핵심 오브젝트 서명이 적용되지 않는 바이트는 특정 *CMD 오브젝트에 따라 다릅니다. 예를 들어, 핵심 서명은 *CMD 오브젝트의 매개변수 디폴트에 적용되지 않습니다. 핵심 오브젝트 서명을 무효화시키지 않는 변경 사항의 예제에는 다음이 포함됩니다.
  - 명령 디폴트 변경.
  - 유효성 검사 프로그램이 없는 명령에 유효성 검사 프로그램 추가.
  - 매개변수 실행이 허용된 위치 변경.
  - 제한된 사용자 매개변수 허용 변경.

서명할 수 있는 iSer ies 오브젝트와 핵심 오브젝트 서명이 적용되는 *CMD 오브젝트의 바이트에 대한 자세한 내용을 알려면 서명 가능한 오브젝트를 참조하십시오.

서명 가능한 오브젝트

오브젝트에 서명하기 위해 사용하는 방법과 상관없이 다양한 OS/400오브젝트 유형을 디지털 서명할 수 있습니다. 라이브러리에 저장된 오브젝트를 제외하고 시스템의 통합 파일 시스템에 저장한 오브젝트(*STMF)를 서명할 수 있습니다. 오브젝트가 첨부된 Java™프로그램일 경우 프로그램도 서명됩니다. QSYS.LIB 파일 시스템의 프로그램(*PGM), 서비스 프로그램(*SRVPGM), 모듈(*MODULE), SQL 패키지(*SQLPKG), *FILE(파일만 저장) 및 명령(*CMD) 오브젝트만 서명할 수 있습니다.

오브젝트에 서명하려면 오브젝트가 로컬 시스템에 상주해야 합니다. 예를 들어, iSeries용 Integrated XSeries Server에서 Windows® 2000 서버를 작동할 경우 통합된 파일 시스템에서 QNTC 파일 시스템을 사용할 수 있습니다. 이 파일 시스템의 디렉토리에는 Windows 2000 오퍼레이팅 시스템에서 소유한 파일이 포함되기 때문에 로컬로 간주되지 않습니다. 빈 오브젝트나 V5R1 이전 릴리스용으로 컴파일된 오브젝트는 서명할 수 없습니다.

명령(*CMD) 오브젝트 서명

*CMD 오브젝트에 서명할 경우 *CMD 오브젝트에 적용할 수 있는 두 가지 서명 유형중 하나를 선택할 수 있습니다. 전체 오브젝트에 서명하거나 오브젝트의 핵심 부분만 서명하기로 선택할 수 있습니다. 전체 오브젝트에 서명하기로 선택한 경우 오브젝트의 비필수 바이트를 제외한 모든 바이트에 서명이 적용됩니다. 전체 오브젝트 서명은 핵심 오브젝트 서명에 들어 있는 항목을 포함합니다.

핵심 오브젝트만 서명하기로 선택한 경우 더 자주 변경되는 바이트는 서명하지 않지만 필수 바이트는 서명에 의해 보호됩니다. 서명되지 않는 바이트는 *CMD 오브젝트에 따라 다르지만, 오브젝트가 유효한 모드를 결정하는 바이트나 오브젝트 실행이 허용된 위치를 결정하는 바이트는 포함시킬 수 있습니다. 예를 들어, 핵심 서명은 *CMD 오브젝트의 매개변수 디폴트에 적용되지 않습니다. 이 유형의 서명을 사용하면 서명을 무효화시키지 않고 명령을 변경할 수 있습니다. 이런 유형의 서명을 무효화시키지 않는 변경 사항의 예제에는 다음이 포함됩니다.

- 명령 디폴트 변경.

- 유효성 검사 프로그램이 없는 명령에 유효성 검사 프로그램 추가.
- 매개변수 실행이 허용된 위치 변경.
- 제한된 사용자 매개변수 허용 변경.

다음 표는 핵심 오브젝트 서명의 일부로 포함된 *CMD 오브젝트의 바이트를 설명합니다.

***CMD 오브젝트의 핵심 오브젝트 서명 구성**

오브젝트의 일부	핵심 오브젝트 서명에 대한 관계
CHGCMDDFT에서 변경한 명령 디폴트	핵심 오브젝트 서명의 일부가 아님
명령 및 라이브러리를 처리하는 프로그램	항상 핵심 오브젝트 서명의 일부로 포함됨
REXX 소스 파일 및 라이브러리	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
REXX 소스 멤버	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
REXX 명령 환경 및 라이브러리	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
REXX 나감 프로그램명, 라이브러리 및 종료 코드	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
유효성 검사 프로그램 및 라이브러리	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
유효한 모드	핵심 오브젝트 서명의 일부가 아님
실행이 허용된 경우	핵심 오브젝트 서명의 일부가 아님
제한된 사용자 허용	핵심 오브젝트 서명의 일부가 아님
도움말 서가	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
도움말 패널 그룹 및 라이브러리	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
도움말 ID	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
도움말 탐색 색인 및 라이브러리	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
현재 라이브러리	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
제품 라이브러리	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
프롬트 대체 프로그램 및 라이브러리	서명할 때 명령에 대해 지정한 경우에는 포함되고, 그렇지 않으면 핵심 오브젝트 서명의 일부가 아님
텍스트(설명)	오브젝트에 저장되어 있기 때문에 핵심 오브젝트 서명이나 전체 오브젝트 서명의 일부가 아님
GUI(Graphical User Interface) 사용	핵심 오브젝트 서명의 일부가 아님

오브젝트 서명 처리

오브젝트에 서명할 경우 오브젝트 서명 처리를 위한 다음 옵션을 지정할 수 있습니다.

- **오류 처리**
  하나 이상의 오브젝트에 서명을 작성한 경우 어플리케이션에서 사용할 오류 처리 유형을 지정할 수 있습니다. 오류가 발생한 경우 어플리케이션에서 오브젝트 서명을 중지하든가 프로세스의 다른 오브젝트 서명을 계속하도록 지정할 수 있습니다.
- **중복된 오브젝트 서명**
  어플리케이션에서 오브젝트를 다시 서명할 경우 어플리케이션에서 서명 프로세스를 처리하는 방법을 지정할 수 있습니다. 원래 서명을 그대로 두거나 원래 서명을 새 서명으로 대체할지 지정할 수 있습니다.
- **서브디렉토리의 오브젝트**
  어플리케이션에서 서브디렉토리에서의 오브젝트 서명을 처리하는 방법을 지정할 수 있습니다. 어플리케이션이 모든 서브디렉토리에서 오브젝트를 개별적으로 서명할지 또는 모든 서브디렉토리를 무시하고 기본 디렉토리 내의 항목만 서명할지 지정할 수 있습니다.
- **오브젝트 서명의 범위**
  *CMD 오브젝트에 서명할 경우 전체 오브젝트에 서명할지 또는 오브젝트의 핵심 부분만 서명할지 지정할 수 있습니다.

서명 확인 처리

서명 확인 처리를 위해 다음 옵션을 지정할 수 있습니다.

- **오류 처리**
  하나 이상의 오브젝트 서명을 확인할 경우 어플리케이션에서 사용할 오류 처리 유형을 지정할 수 있습니다. 오류가 발생한 경우 어플리케이션에서 서명 확인을 중지하든가 프로세스의 다른 오브젝트에서 서명 확인을 계속하도록 지정할 수 있습니다.
- **서브디렉토리의 오브젝트**
  어플리케이션에서 서브디렉토리에서의 오브젝트 서명 확인을 처리하는 방법을 지정할 수 있습니다. 어플리케이션이 모든 서브디렉토리에서 오브젝트의 서명을 개별적으로 확인할지 또는 모든 서브디렉토리를 무시하고 기본 디렉토리 내의 오브젝트에 대한 서명만 확인할지 지정할 수 있습니다.
- **핵심 서명 확인 대 전체 서명 확인**
  확인 프로세스 중에 시스템에서 오브젝트의 핵심 서명과 전체 서명을 처리할 방법을 판별하는 시스템 규칙이 있습니다. 이 규칙은 다음과 같습니다.
  - 오브젝트에 서명이 없을 경우 확인 프로세스는 오브젝트가 서명되지 않았고 프로세스의 다른 오브젝트를 계속 확인할 것을 보고합니다.
  - 오브젝트가 시스템의 신뢰할 수 있는 소스(IBM)에 의해 서명된 경우 서명이 일치해야 합니다. 그렇지 않으면 확인 프로세스에서 오류가 발생합니다. 서명이 일치할 경우 확인 프로세스가 계속됩니다. 서명은 오브젝트의 데이터에 대한 암호화된 수리적인 요약입니다. 그러므로 확인 중 오브젝트의 데이터가 서명했을 때 오브젝트의 데이터와 일치할 경우 서명이 일치하는 것으로 간주됩니다.
  - 오브젝트에 *SIGNATUREVERIFICATION 인증서 저장소에 포함된 인증서를 기준으로 신뢰할 수 있는 전체 오브젝트 서명이 있을 경우 최소한 이 서명 중 하나가 일치해야 합니다. 그렇지 않으면 확인 프로세스에서 오류가 발생합니다. 최소한 하나의 전체 오브젝트 서명이 일치할 경우 확인 프로세스가 계속됩니다.

– 오브젝트에 신뢰할 수 있는 핵심 오브젝트 서명이 있을 경우 최소한 이 서명 중 하나가 *SIGNATUREVERIFICATION 인증서 저장소의 인증서와 일치해야 합니다. 그렇지 않으면 확인 프로세스에서 오류가 발생합니다. 최소한 하나의 핵심 오브젝트 서명이 일치하면 확인 프로세스가 계속됩니다.

오브젝트 서명 및 서명 확인 전제조건

OS/400 오브젝트 서명 및 서명 확인 기능은 iSeries 서버의 오브젝트를 제어할 수 있는 강력한 수단을 추가로 제공합니다. 이 기능을 이용하려면 기능을 사용할 수 있는 전제조건을 충족시켜야 합니다.

오브젝트 서명 전제조건

업무 및 보안 요구사항에 따라 오브젝트에 서명하기 위해 사용할 수 있는 방법이 많습니다.

- DCM(Digital Certificate Manager)을 사용할 수 있습니다.
- Sign Object API를 사용하는 프로그램을 작성할 수 있습니다.
- iSeries 종료점 시스템에 분배하기 위해 오브젝트를 패키지 작성할 때 iSeries Navigator의 중앙 관리 기능을 사용할 수 있습니다.

오브젝트에 서명하기 위해 선택한 방법은 업무 및 보안 요구사항에 따라 달라집니다. 오브젝트에 서명하기 위해 사용하는 방법은 상관없이 다음과 같은 특정한 전제조건을 충족시켜야 합니다.

- DCM(Digital Certificate Manager)을 설치 및 사용하기 위한 전제조건을 충족시켜야 합니다.
  - DCM을 사용하여 *OBJECTSIGNING 인증서 저장소를 작성해야 합니다. 로컬 CA(Certificate Authority) 작성 프로세스의 일부 또는 공용 인터넷 CA로부터 오브젝트 서명 인증서 관리 프로세스의 일부로 이 인증서 저장소를 작성할 수 있습니다.
  - *OBJECTSIGNING 인증서 저장소에는 최소한 하나의 인증서가 포함되어야 하는데, 로컬 CA를 사용하여 작성한 인증서이거나 공용 인터넷 CA로부터 가져온 인증서여야 합니다.
  - DCM을 사용하여 오브젝트 서명에 사용하기 위해 최소한 하나의 오브젝트 서명 어플리케이션 정의를 작성해야 합니다.
  - DCM을 사용하여 오브젝트 서명 어플리케이션 정의에 특정한 인증서를 할당해야 합니다.
- 오브젝트에 서명하는 iSeries 사용자 프로파일에는 *ALLOBJ 특수 권한이 있어야 합니다. *SIGNATUREVERIFICATION 인증서 저장소를 작성하는 iSeries 사용자 프로파일에는 *SECADM 및 *ALLOBJ 특수 권한이 있어야 합니다.

서명 확인 전제조건

다음과 같이 오브젝트의 서명을 확인하기 위해 사용할 수 있는 방법이 많습니다.

- 디지털 인증 관리자(DCM)를 사용할 수 있습니다.
- Verify Object(QYDOVFYO) API를 사용하는 프로그램을 작성할 수 있습니다.
- CHKOBJITG(오브젝트 무결성 검사) 명령과 같은 여러 명령중 하나를 사용할 수 있습니다.

서명 확인을 위해 선택하는 방법은 업무 및 보안 요구사항에 따라 달라집니다. 어떤 방법을 사용하든지에 관계 없이 다음과 같은 특정 전제조건을 충족시켜야 합니다.

- DCM(Digital Certificate Manager)을 설치 및 사용하기 위한 전제조건을 충족시켜야 합니다.
- *SIGNATUREVERIFICATION 인증서 저장소를 작성해야 합니다. 사용자의 요구사항에 따라 두 가지 방법 중 하나로 이 인증서 저장소를 작성할 수 있습니다. 서명 확인 인증서를 관리하기 위해 DCM(Digital Certificate Manager)을 사용하여 인증서 저장소를 작성할 수 있습니다. 또는 공용 인증서를 사용하여 오브젝트에 서명할 경우 Add Verifier(QYDOADDV) API를 사용하는 프로그램을 작성하여 이 인증서 저장소를 작성할 수 있습니다.

  주: Add Verifier API는 디폴트 암호로 인증서 저장소를 작성합니다. DCM을 사용하여 이 디폴트 암호를 인증서 저장소에 대한 권한 없는 액세스를 방지하기 위해 선택한 암호 중 하나로 재설정해야 합니다.
- *SIGNATUREVERIFICATION 인증서 저장소에는 오브젝트를 서명한 인증서의 사본이 포함되어야 합니다. 다음 두 가지 방법 중 하나로 이 인증서를 인증서 저장소에 추가할 수 있습니다. 서명 시스템에서 DCM을 사용하여 인증서를 파일로 내보낸 다음 대상 확인 시스템에서 DCM을 사용하여 인증서를 *SIGNATUREVERIFICATION 인증서 저장소로 가져올 수 있습니다. 또는 공용 인증서를 사용하여 오브젝트에 서명할 경우 Add Verifier API를 사용하는 프로그램을 작성하여 인증서를 대상 확인 시스템의 인증서 저장소에 추가할 수 있습니다.
- *SIGNATUREVERIFICATION 인증서 저장소에는 오브젝트를 서명하는 인증서를 발행한 CA 인증서의 사본이 포함되어야 합니다. 공용 인증서를 사용하여 오브젝트에 서명할 경우 대상 확인 시스템의 인증서 저장소에 필수 CA 인증서의 사본이 있어야 합니다. 그러나 로컬 CA에서 발행한 인증서를 사용하여 오브젝트에 서명할 경우 DCM을 사용하여 로컬 CA 인증서의 사본을 대상 확인 시스템의 인증서 저장소에 추가할 수 있습니다.

  주: 보안상의 이유로 이 Add Verifier API는 CA(Certificate Authority) 인증서를 *SIGNATUREVERIFICATION 인증서 저장소에 넣도록 허용하지 않습니다. CA 인증서를 인증서 저장소에 추가한 경우 시스템은 해당 CA를 신뢰할 수 있는 인증서의 소스로 간주합니다. 따라서 시스템은 이 CA에서 발행한 인증서를 신뢰할 수 있는 소스에서 나온 인증서로서 처리합니다. 그러므로 CA 인증서를 인증서 저장소에 넣기 위해 API를 사용하여 설치 나감 프로그램을 작성할 수 없습니다. CA 인증서를 인증서 저장소에 추가하여 시스템이 신뢰하는 CA를 수동으로 확실하게 제어하려면 디지털 인증 관리자를 사용해야 합니다. 디지털 인증 관리자를 사용하면 시스템이 관리자가 신뢰하지 않은 소스에서 인증서를 가져올 가능성을 방지할 수 있습니다.

  로컬 CA에서 발행한 인증서를 사용하여 오브젝트에 서명할 경우 로컬 CA 호스트 iSeries 서버에서 DCM을 사용하여 로컬 CA 인증서의 사본을 파일로 내보내야 합니다. 그런 다음 대상 확인 iSeries서버에서 DCM을 사용하여 로컬 CA 인증서를 *SIGNATUREVERIFICATION 인증서 저장소에 가져올 수 있습니다. 가능한 오류를 방지하려면 Add Verifier API를 사용하여 서명 확인 인증서를 추가하기 전에 로컬 CA 인증서를 이 인증서 저장소로 가져와야 합니다. 따라서 로컬 CA에서 발행한 인증서를 사용할 경우 DCM을 사용하여 CA 인증서와 확인 인증서를 인증서 저장소로 가져오는 것이 더 쉽다는 것을 알 수 있습니다.

누구도 사용자의 허락 없이 API를 사용하여 확인 인증서를 *SIGNATUREVERIFICATION 인증서 저장소에 추가하지 못하도록 하려면 시스템에서 이 API가 작동하지 않도록 설정하십시오. SST(System Service Tool)를 사용하여 보안 관련 시스템 값의 변경을 허용하지 않으면 이렇게 할 수 있습니다.

- 서명을 확인하는 iSeries 사용자 프로파일에는 *AUDIT 특수 권한이 있어야 합니다. *SIGNATUREERIFICATION 인증서 저장소를 작성하거나 이 인증서 저장소의 암호를 변경하는 iSeries 사용자 프로파일에는 *SECADM 및 *ALLOBJ 특수 권한이 있어야 합니다.

서명된 오브젝트 관리

V5R1부터 IBM에서는 OS/400 서명, 사용권 프로그램 및 PTF를 IBM에서 출시된 오퍼레이팅 시스템을 공식적으로 표시하기 위한 방법 및 시스템 오브젝트에 권한 없는 변경이 있는 경우 이를 감지할 수 있는 방법으로 사용했습니다. 또한 사용자가 구매한 어플리케이션에 협력업체나 다른 공급업체가 서명할 수 있습니다. 따라서 사용자 자신이 직접 오브젝트에 서명하지 않더라도 서명한 오브젝트에 대한 작업 방법과 서명한 오브젝트가 일상적인 시스템 관리 타스크에 미치는 영향을 이해하고 있어야 합니다.

서명된 오브젝트는 기본적으로 백업 및 복원 타스크에 영향을 미치는데, 특히 오브젝트를 시스템에 저장 및 복원하는 방법에 영향을 미칩니다.

서명한 오브젝트에 영향을 미치는 시스템 값과 명령
서명한 오브젝트를 관리하기 위해 사용할 수 있거나 실행했을 때 서명한 오브젝트에 영향을 미치는 시스템 값과 명령에 대해 배웁니다.

서명한 오브젝트에 대한 저장 및 복원 고려 사항
서명한 오브젝트가 시스템에서 저장 및 복원 타스크를 수행하는 방법에 영향을 미치는 것에 대해 배웁니다.

서명 무결성을 확인하기 위한 코드 검사기 명령
오브젝트 무결성을 확인하기 위해 오브젝트 서명을 확인하는 명령 사용에 대한 세부사항을 배웁니다.

서명한 오브젝트에 영향을 미치는 시스템 값과 명령

서명한 오브젝트를 효율적으로 관리하려면 시스템 값과 명령이 서명된 오브젝트에 미치는 영향을 이해해야 합니다. **복원 중 오브젝트 서명 확인**(QVFYOBJRST) 시스템 값에서는 특정한 복원 명령이 서명한 오브젝트에 영향을 미치는 방법과 시스템에서 복원 작업 중에 서명한 오브젝트를 처리하는 방법을 결정합니다. iSeries 시스템에서 서명한 오브젝트를 사용하기 위해 독점적으로 설계된 CL 명령은 없습니다. 그러나 서명한 오브젝트를 관리하거나 오브젝트 서명을 가능하게 하는 인프라 구조 오브젝트를 관리하기 위해 사용하는 일반적인 CL 명령은 많습니다. 다른 명령은 오브젝트에서 서명을 제거하여 서명에서 제공하는 보호 장치를 무효화시킴으로써 시스템에서 서명한 오브젝트에 부정적인 영향을 미칠 수 있습니다.

서명한 오브젝트에 영향을 미치는 시스템 값

OS/400시스템 값의 복원 범주 멤버인 **복원 중 오브젝트 서명 확인**(QVFYOBJRST) 시스템 값은 명령이 시스템의 서명한 오브젝트에 영향을 미치는 방법을 결정합니다. iSeries Navigator를 통해 사용할 수 있는 이 시스

템 값은 복원 작업 중에 시스템에서 서명 확인을 처리하는 방법을 제어합니다. 다른 두 시스템 값 설정과 함께 이 시스템 값에 사용하는 설정은 시스템의 복원 작업에 영향을 미칩니다. 이 값을 위해 선택한 설정에 따라 오브젝트의 서명 상태를 기준으로 오브젝트가 복원되는 것을 허용하거나 허용하지 않을 수 있습니다. 예를 들어, 오브젝트가 서명되지 않았는지, 오브젝트에 잘못된 구조가 있는지, 신뢰할 수 있는 소스에서 서명했는지 등입니다. 이 시스템 값의 기본 설정을 사용하면 서명되지 않은 오브젝트를 복원할 수 있습니다. 그러나 오브젝트에 유효한 서명이 있을 경우에만 서명한 오브젝트를 복원할 수 있습니다. 오브젝트에 시스템에서 신뢰하는 서명이 있는 경우에만 시스템에서 오브젝트를 서명한 오브젝트로 정의합니다. 시스템은 오브젝트의 다른 ″신뢰할 수 없는″ 서명을 무시하고 오브젝트가 서명되지 않은 것으로 처리합니다.

모든 서명을 무시하는 값에서부터 시스템에서 복원하는 모든 오브젝트의 유효한 서명을 요구하는 값에 이르기까지 QYFYOBJRST 시스템 값에 사용할 수 있는 여러 값이 있습니다. 이 시스템 값은 프로그램(*PGM), 명령(*CMD), 서비스 프로그램(*SRVPGM), SQL 패키지(*SQLPKG) 및 모듈(*MODULE)과 같이 복원할 실행 가능한 오브젝트에만 영향을 미칩니다. 또한 CRTJVAPGM(Java 프로그램 작성) 명령으로 작성된 연관된 Java 프로그램이 있는 스트림 파일(*STMF) 오브젝트에도 적용됩니다. 파일 저장(*SAV)이나 IFS 파일에는 적용되지 않습니다.

이 시스템 값과 다른 시스템 값에 대해 더 자세히 배우려면 Information Center의 System Value Finder를 참조하십시오.

서명한 오브젝트에 영향을 미치는 CL 명령

서명한 오브젝트와 같이 사용하거나 iSeries 서버의 서명한 오브젝트에 영향을 미치는 여러 CL 명령이 있습니다. 다양한 명령을 사용하여 오브젝트의 서명 정보를 확인하고, 오브젝트의 서명을 확인하고, 서명을 확인하는 데 필요한 보안 오브젝트를 저장 및 복원할 수 있습니다. 그리고 실행할 경우 오브젝트에서 서명을 제거하고 서명에서 제공하는 보안을 무효화시키는 명령 그룹이 있습니다.

오브젝트의 서명 정보를 보기 위한 명령

- 오브젝트 설명 표시(DSPOBJD) 명령.
  이 명령은 지정한 라이브러리나 스레드의 라이브러리 리스트의 라이브러리에 있는 지정한 오브젝트의 이름과 속성을 보여줍니다. 이 명령을 사용하여 오브젝트에 서명했는지 여부를 확인하고 서명에 대한 정보를 볼 수 있습니다.
- 오브젝트 링크 표시(DSPLNK) 및 오브젝트 링크에 대한 작업(WRKLNK) 통합 파일 시스템 명령.
  이 명령을 사용하여 통합된 파일 시스템의 오브젝트에 대한 서명 정보를 표시할 수 있습니다.

오브젝트 서명을 확인하기 위한 명령

- 오브젝트 무결성 검사(CHKOBJITG) 명령.
  이 명령을 사용하면 시스템의 오브젝트에 무결성 위반이 있는지 확인할 수 있습니다. 바이러스가 파일이나 시스템의 다른 오브젝트를 손상시킨 경우 바이러스 확인 프로그램을 사용하여 확인하는 것과 같은 방법으로 이 명령을 사용하여 서명을 확인할 수 있습니다. 서명한 오브젝트 및 서명 가능한 오브젝트와 함께 이 명령의 사용에 대해 더 자세히 배우려면 서명 무결성을 보장하는 코드 검사기 명령을 참조하십시오.

- 제품 옵션 검사(CHKPRDOPT) 명령.
  이 명령은 소프트웨어 제품의 올바른 구조와 실제 구조 사이의 차이점을 보고합니다. 예를 들어, 설치된 제품에서 오브젝트를 삭제하면 명령에서 오류를 보고합니다. CHKSIG 매개변수를 사용하여 명령에서 제품에 대해 가능한 서명 문제를 처리 및 보고하는 방법을 지정할 수 있습니다. 서명한 오브젝트와 서명 가능한 오브젝트와 함께 이 명령을 사용하는 것에 대해 더 자세히 배우려면 서명 무결성을 보장하는 코드 검사기 명령을 참조하십시오.
- 사용권 프로그램 저장(SAVLICPGM) 명령.
  이 명령은 사용권 프로그램을 구성하는 오브젝트의 사본을 저장합니다. RSTLICPGM(사용권 프로그램 복원) 명령으로 복원할 수 있는 양식으로 사용권 프로그램을 저장합니다. CHKSIG 매개변수를 사용하여 명령에서 제품에 대해 가능한 서명 문제를 처리 및 보고하는 방법을 지정할 수 있습니다. 서명한 오브젝트 및 서명 가능한 오브젝트와 함께 이 명령을 사용하는 것에 대해 더 자세히 배우려면 서명 무결성을 보장하는 코드 검사기 명령을 참조하십시오.
- 복원(RST) 명령.
  이 명령은 통합된 파일 시스템(IFS)에서 사용할 수 있는 하나 이상의 오브젝트 사본을 복원합니다. 이 명령을 사용하여 인증서 저장소와 그 내용을 시스템에 복원할 수도 있습니다. 그러나 이 명령을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 복원할 수 없습니다. 복원 명령에서 서명된 오브젝트와 서명 가능한 오브젝트를 처리하는 방법은 복원 중 오브젝트 서명 확인(QVFYOBJRST) 시스템 값에 설정에 의해 결정됩니다.
- 라이브러리 복원(RSTLIB) 명령.
  이 명령 SAVLIB(라이브러리 저장) 명령으로 저장한 라이브러리 하나 또는 라이브러리 그룹을 복원합니다. RSTLIB 명령은 라이브러리 설명, 오브젝트 설명, 라이브러리의 오브젝트 내용을 포함하는 전체 라이브러리를 복원합니다. 이 명령에서 서명된 오브젝트와 서명 가능한 오브젝트를 처리하는 방법은 복원 중 오브젝트 서명 확인(QVFYOBJRST) 시스템 값의 설정에 의해 결정됩니다.
- 사용권 프로그램 복원(RSTLICPGM) 명령.
  이 명령은 초기 설치 또는 신규 릴리스 설치를 위해 사용권 프로그램을 로드 또는 복원합니다. 이 명령에서 서명된 오브젝트와 서명 가능한 오브젝트를 처리하는 방법은 복원 중 오브젝트 서명 확인(QVFYOBJRST) 시스템 값에 의해 결정됩니다.
- 오브젝트 복원(RSTOBJ) 명령.
  이 명령은 디스켓, 테이프, 광 볼륨에 저장된 하나의 라이브러리나 하나의 명령을 사용한 저장 파일에서 하나 이상의 오브젝트를 복원합니다. 이 명령에서 서명된 오브젝트와 서명 가능한 오브젝트를 처리하는 방법은 복원 중 오브젝트 서명 확인(QVFYOBJRST) 시스템 값의 설정에 의해 결정됩니다.

인증서 저장소를 저장 및 복원하기 위한 명령

- 저장(SAV) 명령.
  이 명령을 사용하면 인증서 저장소를 포함하여 통합된 파일 시스템에서 사용할 수 있는 하나 이상의 오브젝트 사본을 저장합니다. 그러나 이 명령을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 복원할 수 없습니다.
- 보안 자료 저장(SAVSECDTA) 명령.
  이 명령을 사용하면 시스템을 제한된 상태로 두지 않고도 모든 보안 정보를 저장할 수 있습니다. 이 명령을

사용하면 *SIGNATUREVERIFICATION 인증서 저장소와 거기에 포함된 인증서를 저장할 수 있습니다. 이 명령에서 다른 인증서 저장소는 저장하지 않습니다.

- 시스템 저장(SAVSYS) 명령.
  이 명령을 사용하면 사용권 내부 코드 사본과 QSYS 라이브러리를 iSeries 서버 설치와 호환되는 형식으로 저장할 수 있습니다. 다른 라이브러리의 오브젝트는 저장하지 않습니다. 그리고 이 명령을 사용하면 SAVSECDTA 명령과 SAVCFG 명령을 사용하여 저장할 수 있는 보안 및 구성 오브젝트를 저장할 수 있습니다. 이 명령을 사용하면 *SIGNATUREVERIFICATION 인증서 저장소와 해당 저장소에 포함된 인증서를 저장할 수 있습니다.
- 복원(RST) 명령.
  이 명령을 사용하면 인증서 저장소와 그 내용을 시스템에 복원할 수 있습니다. 그러나 이 명령을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소를 복원할 수 없습니다.
- 사용자 프로파일 복원(RSTUSRPRF) 명령.
  이 명령을 사용하면 SAVSYS(시스템 저장) 명령이나 SAVSECDTA(보안 자료 저장) 명령으로 저장한 사용자 프로파일의 기본적인 부분이나 사용자 프로파일 세트를 복원할 수 있습니다. 이 명령을 사용하여 *SIGNATUREVERIFICATION 인증서 저장소와 이 인증서 저장소를 포함한 다른 모든 인증서 저장소의 은닉된 암호를 복원할 수 있습니다. 사용자 프로파일 정보를 복원하지 않고도 SECDTA 매개변수의 값으로 *DCM을 지정하고 USRPRF 매개변수에 대해 *NONE를 지정하여 *SIGNATUREVERIFICATION 인증서 저장소를 복원할 수 있습니다. 이 명령을 사용하여 사용자 프로파일 정보와 인증서 저장소 및 해당 암호를 복원하려면 USRPRF 매개변수에 대해 *ALL을 지정합니다.

오브젝트에서 서명을 제거할 수 있는 명령

서명된 오브젝트에서 다음 명령을 사용하면 오브젝트에서 서명을 제거하는 방법으로 서명을 제거할 수 있습니다. 서명을 제거하면 영향을 받는 오브젝트에 문제가 생길 수 있습니다. 최소한 오브젝트의 소스가 신뢰할 수 있는 소스인지 확인할 수 없고 서명을 확인하여 오브젝트의 변경 사항을 감지할 수 없습니다. IBM이나 공급업체 같이 다른 사용자에게서 가져온 서명된 오브젝트와는 반대로 이 명령은 사용자가 작성한 서명된 오브젝트에만 사용해야 합니다. 명령으로 오브젝트의 서명이 제거되었는지 궁금하면 DSPOBJD(오브젝트 설명 표시) 명령을 사용하여 서명이 있는지 확인하고 필요하면 다시 서명할 수 있습니다.

주: 저장 명령으로 오브젝트의 서명이 없어졌는지 확인하려면 오브젝트를 저장한 라이브러리 외의 다른 라이브러리(예: QTEMP)로 오브젝트를 복원해야 합니다. 그런 다음 DSPOBJD 명령을 사용하여 저장 매체의 오브젝트에서 서명이 제거되었는지 확인하십시오.

- 프로그램 변경(CHGPGM) 명령.
  이 명령은 프로그램을 다시 컴파일하지 않고도 프로그램의 속성을 변경합니다. 또한 지정한 속성이 현재 속성과 같은 경우에도 이 명령을 사용하여 프로그램을 다시 작성할 수 있습니다.
- 서비스 프로그램 변경(CHGSRVPGM) 명령.
  이 명령은 프로그램을 다시 컴파일하지 않고도 서비스 프로그램의 속성을 변경합니다. 또한 지정한 속성이 현재 속성과 같은 경우에도 이 명령을 사용하여 서비스 프로그램을 다시 작성할 수 있습니다.

- 저장 파일 지우기(CLRSAVF) 명령.
  이 명령은 저장 파일의 내용을 지웁니다. 저장 파일의 기존 레코드를 모두 지우고 파일에서 사용하는 기억 장치 양을 줄입니다.
- 저장(SAV) 명령.
  이 명령은 통합된 파일 시스템에서 사용할 수 있는 하나 이상의 오브젝트 사본을 저장합니다. 이 명령을 사용할 경우 TGTRLS 매개변수에 대해 V5R2M0 이전의 값을 지정하면 저장 매체의 명령(*CMD) 오브제트에서 서명을 제거할 수 있습니다. V5R2 이전 릴리스에서는 명령 오브젝트를 서명할 수 없기 때문에 서명이 제거됩니다.
- 라이브러리 저장(SAVLIB) 명령.
  이 명령을 사용하면 하나 이상의 라이브러리 사본을 저장할 수 있습니다. 이 명령을 사용할 경우 TGTRLS 매개변수에 대해 V5R2M0 이전의 값을 지정하면 저장 매체의 명령(*CMD) 오브제트에서 서명을 제거할 수 있습니다. V5R2 이전 릴리스에서는 명령 오브젝트를 서명할 수 없기 때문에 서명이 제거됩니다.
- 오브젝트 저장(SAVOBJ) 명령.
  이 명령은 같은 라이브러리에 있는 하나의 오브젝트나 오브젝트 그룹을 저장합니다. 이 명령을 사용할 경우 TGTRLS 매개변수에 대해 V5R2M0 이전의 값을 지정하면 저장 매체의 명령(*CMD) 오브젝트에서 서명을 제거할 수 있습니다. V5R2 이전 릴리스에서는 명령 오브젝트를 서명할 수 없기 때문에 서명이 제거됩니다.

서명한 오브젝트에 대한 저장 및 복원 고려 사항

iSeries 서버의 복원 작업에 영향을 미칠 수 있는 여러 시스템 값이 있습니다. 이 시스템 값 중에서 **복원 중 오브젝트 서명 확인(QVFYOBJRST)** 시스템 값에서만 서명된 오브젝트를 복원할 때 시스템에서 서명한 오브젝트를 처리하는 방법을 결정합니다. 이 시스템 값에 대해 선택한 설정을 사용하면 복원 프로세스에서 서명이 없는 오브젝트나 유효하지 않은 서명이 있는 오브젝트를 확인하는 방법을 결정할 수 있습니다.

이루 저장 및 복원 명령은 서명한 오브젝트에 영향을 미치고, 저장 및 복원 작업 중에 서명한 오브젝트와 서명하지 않은 오브젝트를 시스템에서 처리하는 방법을 결정합니다. 이 명령과 이 명령이 서명한 오브젝트에 미치는 영향을 알고 있으면 시스템을 더 잘 관리하고 발생할 수 있는 잠재적인 문제를 피할 수 있음을 알고 있어야 합니다.

다음 명령을 사용하면 저장 및 복원 작업 중에 오브젝트의 서명을 확인할 수 있습니다.

- 사용권 프로그램 저장(SAVLICPGM) 명령.
- 복원(RST) 명령.
- 라이브러리 복원(RSTLIB) 명령.
- 사용권 프로그램 복원(RSTLICPGM) 명령.
- 오브젝트 복원(RSTOBJ) 명령.

이 명령을 사용하면 인증서 저장소를 저장하고 복원할 수 있습니다. 인증서 저장소는 오브젝트에 서명하고 서명을 확인하기 위해 사용한 인증서가 포함되어 있는 보안에 민감한 오브젝트입니다.

- 저장(SAV) 명령.

- 보안 자료 저장(SAVSECDTA) 명령.
- 시스템 저장(SAVSYS) 명령.
- 복원(RST) 명령.
- 사용자 프로파일 복원(RSTUSRPRF) 명령.

사용한 매개변수 값에 따라 일부 저장 명령은 서명에서 제공하는 보안을 무효화하여 저장 매체의 오브젝트에서 서명을 제거할 수 있습니다. 예를 들어, V5R2MO 이전의 대상 릴리스로 명령(*CMD) 오브젝트를 참조하는 모든 저장 작업에서는 서명없이 명령을 저장합니다. 서명을 제거하면 영향을 받는 오브젝트에 문제가 생길 수 있습니다. 최소한 오브젝트의 소스가 신뢰할 수 있는 소스인지 확인할 수 없고 서명을 확인하여 오브젝트의 변경 사항을 감지할 수 없습니다. IBM이나 공급업체 같이 다른 사용자에게서 가져온 서명된 오브젝트와는 반대로 이 명령은 사용자가 작성한 서명된 오브젝트에만 사용해야 합니다.

주: 저장 명령으로 오브젝트의 서명이 없어졌는지 확인하려면 오브젝트를 저장한 라이브러리 외의 다른 라이브러리(예: QTEMP)로 오브젝트를 복원해야 합니다. 그런 다음 DSPOBJD 명령을 사용하여 저장 매체의 오브젝트에서 서명이 제거되었는지 확인하십시오.

일반적인 저장 명령 뿐만 아니라 다음과 같은 특정 저장 명령에 대해서도 이러한 잠재성을 알고 있어야 합니다.

- 저장(SAV) 명령.
- 라이브러리 저장(SAVLIB) 명령.
- 오브젝트 저장(SAVOBJ) 명령.

저장 및 복원 작업 중에 이 명령이 서명한 오브젝트와 오브젝트 서명에 영향을 미치는 방법에 대한 더 자세한 내용은 서명한 오브젝트에 영향을 미치는 시스템 값과 명령을 참조하십시오.

서명 무결성을 확인하기 위한 코드 검사기 명령

DCM(Digital Certificate Manager)이나 API를 사용하여 오브젝트의 서명을 확인할 수 있습니다. 또한 여러 명령을 사용하여 서명을 확인할 수 있습니다. 이러한 명령을 사용하면 바이러스가 시스템의 파일이나 다른 오브젝트를 손상시킨 경우 바이러스 검사 프로그램을 사용하여 확인하는 것과 같은 방법으로 서명을 확인할 수 있습니다. 대부분의 서명은 오브젝트를 시스템에 복원 또는 설치할 때 검사합니다. 예를 들어, RSTLIB 명령을 사용하여 검사합니다.

세 가지 명령 중 하나를 선택하여 이미 시스템에 있는 오브젝트의 서명을 검사할 수 있습니다. 그 중에서 CHKOBJITG(오브젝트 무결성 검사) 명령은 특히 오브젝트 서명을 확인하기 위해 설계되었습니다. 각 명령의 서명 검사는 CHKSIG 매개변수로 제어됩니다. 이 매개변수를 사용하면 서명할 수 있는 모든 오브젝트 유형을 확인하거나, 모든 서명을 무시하거나, 서명이 있는 오브젝트만 검사할 수 있습니다. 이 마지막 옵션은 매개변수의 디폴트 값입니다.

CHKOBJITG(오브젝트 무결성 검사) 명령

오브젝트 무결성 검사(CHKOBJITG) 명령을 사용하면 시스템의 오브젝트에 무결성 위반이 있는지 확인할 수 있습니다. 이 명령을 사용하여 특정한 사용자 프로파일이 소유한 오브젝트, 특정한 경로명과 일치하는 오브젝트 또는 시스템의 모든 오브젝트에 대해 무결성 위반을 검사할 수 있습니다. 다음 조건 중 하나가 충족될 경우 무결성 위반 기록부 항목이 발생합니다.

- 명령, 프로그램, 모듈 오브젝트 또는 라이브러리의 속성이 변경되었습니다.
- 오브젝트의 디지털 서명이 유효하지 않은 것으로 결정되었습니다. 서명은 오브젝트의 데이터에 대한 암호화된 수리적인 요약입니다. 그러므로 확인 중 오브젝트의 데이터가 서명했을 때 오브젝트의 데이터와 일치할 경우 서명이 일치되고 유효한 것으로 간주됩니다. 오브젝트를 서명할 때 작성된 암호화된 수리적인 요약과 서명 확인 중에 작성된 암호화된 수리적인 요약을 비교하여 유효하지 않은 서명이 결정됩니다. 서명 확인 프로세스는 두 가지 요약 값을 비교합니다. 값이 같지 않으면 오브젝트를 서명한 후 오브젝트 내용이 변경된 것이고 서명은 유효하지 않은 것으로 간주됩니다.
- 오브젝트에 오브젝트 유형에 대한 잘못된 정의역 속성이 있습니다.

명령에서 오브젝트의 무결성 위반을 발견한 경우 오브젝트명, 라이브러리명(또는 경로명), 오브젝트 유형, 오브젝트 소유자 및 실패 유형 등이 데이터베이스 기록부 파일에 추가됩니다. 무결성 위반은 아니지만 특정한 다른 경우에도 기록부 항목을 만듭니다. 예를 들어, 서명할 수 있지만 디지털 서명이 없는 오브젝트, 검사할 수 없는 오브젝트, 현재 시스템 구현에서 사용하려면 형식을 바꾸어야 하는 오브젝트의 경우 이 명령에서 기록부 항목을 만듭니다.

CHKSIG 매개변수 값은 명령에서 오브젝트의 디지털 서명을 처리하는 방법을 제어합니다. 이 매개변수에 다음 세 가지 값 중 하나를 지정할 수 있습니다.

- *SIGNED - 이 값을 지정하면 명령에서 디지털 서명이 있는 오브젝트를 검사합니다. 유효하지 않은 서명이 있는 오브젝트에 대해 기록부 항목을 만듭니다. 이것이 디폴트 값입니다.
- *ALL - 이 값을 지정하면 명령에서 모든 서명 가능한 오브젝트를 검사하여 서명이 있는지 확인합니다. 서명이 없는 서명 가능한 오브젝트와 유효하지 않은 서명이 있는 오브젝트에 대해 기록부 항목을 만듭니다.
- *NONE - 이 값을 지정하면 명령에서 오브젝트의 디지털 서명을 검사하지 않습니다.

CHKPRDOPT(제품 옵션 확인) 명령

제품 옵션 확인(CHKPRDOPT) 명령은 소프트웨어 제품의 올바른 구조와 실제 구조 사이의 차이점을 보고합니다. 예를 들어, 설치된 제품에서 오브젝트를 삭제하면 명령이 오류를 보고합니다.

CHKSIG 매개변수 값은 명령에서 오브젝트의 디지털 서명을 처리하는 방법을 제어합니다. 이 매개변수에 다음 세 가지 값 중 하나를 지정할 수 있습니다.

- *SIGNED - 이 값을 지정하면 명령에서 디지털 서명이 있는 오브젝트를 검사합니다. 명령에서 서명한 오브젝트의 서명을 확인합니다. 오브젝트의 서명이 유효하지 않은 것으로 결정되면 작업 기록부에 메세지를 보내고 오류 상태인 제품을 식별합니다. 이것이 디폴트 값입니다.
- *ALL - 이 값을 지정하면 명령에서 모든 서명 가능한 오브젝트를 검사하여 서명이 있는지 확인하고 오브젝트의 서명을 검사합니다. 서명이 없는 서명 가능한 오브젝트에 대해서는 작업 기록부에 메세지를 보냅니

다. 그러나 명령에서 제품을 오류로 판별하지 않습니다. 오브젝트의 서명이 유효하지 않은 것으로 결정되면 작업 기록부에 메세지를 보내고 제품을 오류로 설정합니다.

- *NONE - 이 값을 지정하면 명령에서 제품 오브젝트의 디지털 서명을 검사하지 않습니다.

SAVLICPGM(사용권 프로그램 저장) 명령

사용권 프로그램 저장(SAVLICPGM) 명령을 사용하면 사용권 프로그램을 구성하는 오브젝트의 사본을 저장할 수 있습니다. RSTLICPGM(사용권 프로그램 복원) 명령으로 복원할 수 있는 양식으로 사용권 프로그램을 저장합니다.

CHKSIG 매개변수 값은 명령에서 오브젝트의 디지털 서명을 처리하는 방법을 제어합니다. 이 매개변수에 다음 세 가지 값 중 하나를 지정할 수 있습니다.

- *SIGNED - 이 값을 지정하면 명령에서 디지털 서명이 있는 오브젝트를 검사합니다. 서명한 오브젝트의 서명은 검사하지만 서명하지 않은 오브젝트는 검사하지 않습니다. 오브젝트의 서명이 유효하지 않은 것으로 결정되면 작업 기록부에 오브젝트를 식별하는 메세지를 보내고 저장되지 않습니다. 이것이 디폴트 값입니다.
- *ALL - 이 값을 지정하면 명령에서 모든 서명 가능한 오브젝트를 검사하여 서명이 있는지 확인하고 오브젝트의 서명을 검사합니다. 서명이 없는 서명 가능한 오브젝트에 대해서는 작업 기록부에 메세지를 보냅니다. 그러나 저장 프로세스는 종료되지 않습니다. 오브젝트의 서명이 유효하지 않은 것으로 결정되면 작업 기록부에 메세지를 보내고 저장되지 않습니다.
- *NONE - 이 값을 지정하면 명령에서 제품 오브젝트의 디지털 서명을 검사하지 않습니다.

서명한 오브젝트의 문제 해결

다음 표를 사용하면 iSeries 오브젝트 서명 기능과 서명 확인 기능을 사용할 때 발생할 수 있는 일반적인 문제의 해결을 도와주는 정보를 찾을 수 있습니다.

일반적인 오브젝트 서명 문제점

문제	가능한 솔루션
V4R5 이전 대상 릴리스로 Sign Object API를 사용하여 오브젝트에 서명할 경우 서명 프로세스에서 오류가 발생하고 오브젝트는 서명되지 않습니다(오류 메시지 CPFB721).	iSeries에서는 V5R1까지 오브젝트 서명을 지원하지 않습니다. CPFB721 오류 메시지를 리턴하는 오브젝트의 경우 오브젝트에 서명을 하려면 V5R1 이상의 대상 릴리스로 프로그램을 다시 작성해야 합니다.

일반적인 서명 확인 문제점

문제	가능한 솔루션
서명 없는 오브젝트에 대한 복원 프로세스가 실패했습니다.	서명이 없는 것이 문제가 되지 않으면 QVFYOBJRST 시스템 값을 5로 설정했는지 확인하십시오. 값 5는 서명하지 않은 오브젝트를 복원할 수 없도록 지정합니다. 값을 3으로 바꾸고 다시 복원해 보십시오.

문제	가능한 솔루션
서명 있는 오브젝트에 대한 복원 프로세스가 실패했습니다.	*SIGNATUREVERIFICATION 인증서 저장소를 시스템으로 전송하고 DCM을 사용하여 암호를 변경하지 않은 경우 이 문제가 발생할 수 있습니다. 이런 경우 저장소에 들어 있는 인증서를 사용하여 복원 프로세스 중에 오브젝트의 서명을 확인할 수 없습니다. DCM을 사용하여 인증서 저장소의 암호를 변경하십시오. 암호를 모르면 인증서 저장소를 삭제해야 합니다. 그리고 인증서 저장소를 다시 작성한 다음 DCM을 사용하여 암호를 변경하십시오.
제품을 복원하거나 설치할 경우 서명 검사가 실패하면 오류가 발생합니다.	오브젝트 서명 검사가 제대로 되지 않으면 이 오류는 오브젝트를 서명한 후 오브젝트가 변경되었음을 나타냅니다. 오브젝트 무결성이 문제일 경우 QVFYOBJRST 시스템 값을 변경하지 말거나 해당하는 오브젝트를 복원할 수 있게 해주는 다른 조치를 수행하십시오. 그렇게 하면 서명 확인에서 제공하는 보안을 피해가거나 해로운 오브젝트가 시스템에 허용될 수 있습니다. 대신 오브젝트 서명자에게 문의하여 문제를 해결하기 위해 취해야 할 적절할 조치를 확인하십시오.

오브젝트 서명 및 서명 확인에 관련된 정보

오브젝트 서명 및 서명 확인 기능은 비교적 새로운 보안 기술입니다. 다음은 이런 기술과 사용 방법에 대해 더 알고 싶을 경우 도움이 될 수 있는 다른 자원 리스트입니다.

- **VeriSign Help Desk 웹 사이트**
  VeriSign 웹 사이트에서는 많은 다른 인터넷 보안 주제 뿐만 아니라 오브젝트 서명과 같은 디지털 인증에 대한 광범위한 라이브러리를 제공합니다.
- **IBM eServer iSeries Wired Network Security: OS/400 V5R1 DCM and Cryptographic Enhancements SG24-6168**
  이 IBM 레드북은 V5R1 네트워크 보안 확장 기능에 중점을 둡니다. 레드북에서는 iSeries 오브젝트 서명 기능, DCM(Digital Certificate Manager) 등을 사용하는 방법을 포함하여 많은 주제를 설명합니다.

Printed in U.S.A