



@server

iSeries

Podepisování objektů a ověření podpisu







@server

iSeries

Podepisování objektů a ověření podpisu



---

# Obsah

<b>Podepisování objektů a ověření podpisu</b> . . . . .	1
Co je nového ve verzi V5R2 . . . . .	2
Tisk tohoto tématu. . . . .	3
Scénáře podepisování objektů . . . . .	3
Scénář: Použití produktu DCM (Digital Certificate Manager) k podepisování objektů a ověřování podpisů . . . . .	4
Podrobnosti konfigurace . . . . .	8
Scénář: Použití rozhraní API k podepisování objektů a ověřování podpisů . . . . .	13
Podrobnosti konfigurace . . . . .	18
Scénář: Použití Centrální správy k podepisování objektů . . . . .	25
Podrobnosti konfigurace . . . . .	29
Koncepce podepisování objektů . . . . .	33
Digitální podpisy . . . . .	34
Podepisovatelné objekty . . . . .	35
Zpracování podepisování objektů . . . . .	36
Zpracování ověřování podpisů . . . . .	37
Nezbytné předpoklady pro podepisování objektů a ověřování podpisů . . . . .	37
Správa podepsaných objektů . . . . .	39
Systémové hodnoty a příkazy, které ovlivňují podepsané objekty . . . . .	40
Pokyny pro ukládání a obnovu podepsaných objektů . . . . .	43
Příkazy programu pro kontrolu kódu k zajištění integrity podpisu . . . . .	44
Odstraňování problémů s podepsanými objekty . . . . .	45
Související informace pro podepisování objektů a ověření podpisů . . . . .	46



---

# Podepisování objektů a ověření podpisu

Podepisování objektů a ověření podpisu jsou schopnosti zabezpečení, které můžete využít při ověřování integrity různých objektů iSeries. Pomocí soukromého klíče digitálního certifikátu podepíšete objekt a pomocí certifikátu (který obsahuje odpovídající veřejný klíč) ověříte digitální podpis. Digitální podpis zajišťuje integritu času a obsahu objektu, který jste podepsali. Podpis je nepopíratelný důkaz pravosti i autorizace. Je možné ho použít k prokázání původu a k detekování falšování. Podepsáním objektu identifikujete zdroj objektu a poskytnete prostředek, pomocí kterého je možné určit změny objektu. Po ověření podpisu na objektu můžete určit, zda byly provedeny změny v jeho obsahu od okamžiku jeho podepsání. Můžete také ověřit zdroj podpisu, abyste se přesvědčili o hodnověrnosti původu objektu.

Podepisování objektů a ověření podpisu můžete na serveru iSeries implementovat pomocí:

- Rozhraní API, chcete-li programově podepisovat objekty a ověřovat podpisy na objektech.
- Produktu DCM (Digital Certificate Manager), chcete-li podepisovat objekty a prohlížet nebo ověřovat podpisy.
- Centrální správy v prostředí produktu iSeriesNavigator, chcete-li podepisovat objekty jako součást distribučních programových balíčků, určených pro použití jinými systémy.
- Pomocí CL příkazů, jako např. CHKOBJITG (Check Object Integrity), chcete-li ověřovat podpisy.

Pokud chcete získat více informací o výše uvedených metodách podepisování objektů a o tom, jak podepisování objektů může obohatit vaši aktuální strategii zabezpečení ochrany dat, prostudujte si následující části:

## Co je nového ve verzi V5R2

Zde získáte informace o nových schopnostech iSeries, pokud jde o podepisování objektů a ověření podpisu, a také o změnách v dokumentaci pro toto vydání.

## Tisk tohoto tématu

Pomocí těchto informací vytisknete celé téma jako soubor ve formátu PDF.

## Scénáře podepisování objektů

Uvedené informace použijte k prohlédnutí scénářů, které představují některé typické situace využití schopností iSeries v oblasti podepisování objektů a ověření podpisu. U každého scénáře jsou také uvedeny všechny konfigurační úlohy, které musíte provést, aby bylo možné scénář použít tak, jak je zde popsán.

## Koncepce podepisování objektů

Pomocí těchto koncepcí a referenčních informací získáte znalosti o digitálních podpisech a o tom, jak fungují procesy podepisování objektů a ověření podpisu.

## Předpoklady pro použití podepisování objektů a ověření podpisu

V této části se dozvíte informace o nezbytných předpokladech pro konfiguraci a také další požadavky, které je třeba vzít v úvahu při plánování použití podepisování objektů a ověření podpisu.

## Správa podepsaných objektů

Zde naleznete informace o příkazech a systémových hodnotách iSeries, které budete potřebovat pro práci s podepsanými objekty, a informace o tom, jakým způsobem podepsané objekty ovlivňují procesy zálohování a obnovy dat.

## Řešení problémů s podepisováním objektů a ověřením podpisu

V této části naleznete informace, jak vyřešit problémy a chyby, se kterými se můžete setkat při používání podepisování objektů a ověřování podpisů.

## Související informace pro podepisování objektů a ověření podpisu

Tyto informace obsahují odkazy na jiné zdroje, které obsahují další informace o podepisování objektů a ověřování podpisů objektu.

---

## Co je nového ve verzi V5R2

Schopnosti podepisování objektů a ověření podpisu pro iSeries byly poprvé představeny ve verzi V5R1. Ve verzi V5R2 byly k těmto schopnostem přidány některé další funkce a vylepšení.

Nové nebo vylepšené funkce podepisování objektů a ověření podpisu zahrnují:

- **Funkci podepisování objektů v Centrální správě v prostředí produktu iSeries Navigator**  
Nyní můžete pomocí průvodce definicí produktů v Centrální správě podepisovat objekty, které balíte za účelem distribuce na koncové systémy iSeries.
- **Podepisování objektů typu command (\*CMD)**  
Nově můžete podepisovat i objekty \*CMD. Můžete zvolit, zda se má podepisovat celý objekt \*CMD, nebo zda se mají podepsat pouze komponenty jádra objektu \*CMD.
- **Nová rozhraní API pro podepisování a ověření**  
Můžete používat tři rozhraní API, pomocí kterých můžete plně využít výhod vylepšení schopností podepisování a ověření systému OS/400:
  - Rozhraní Sign Buffer (QYDOSGNB, QydoSignBuffer) API  
Toto rozhraní API dovoluje lokálnímu systému, aby digitálně podepsal vyrovnávací paměť. Tímto způsobem osvědčí, že je důvěryhodná. Po podepsání vyrovnávací paměti systém vrátí digitální podpis tomu, kdo vyvolal API. Toto rozhraní API můžete například používat, když chcete podepsat část XML souboru a uložit podpis do jiné části souboru ve formátu XML. Nebo můžete číst záznamy databázového souboru do vyrovnávací paměti a použít rozhraní API k jejich podpisu.
  - Rozhraní Verify Buffer (QYDOVFYB, QydoVerifyBuffer) API  
Toto rozhraní API umožňuje lokálnímu systému ověřit digitální podpis na dříve podepsané vyrovnávací paměti.
  - Rozhraní Add Verifier (QYDOADDV, QydoAddVerifier) API  
Toto rozhraní API přidává certifikát do paměti certifikátů \*SIGNATUREVERIFICATION systému. Systém pak může použít přidání certifikát k ověření podpisů na objektech, které certifikát vytvořil. Ověřování podpisu umožňuje systému ověřit integritu podepsaných objektů, aby bylo zajištěno, že objekty nebyly změněny od doby jejich podpisu. Pokud neexistuje paměť certifikátů, uvedené rozhraní API ji vytvoří, jakmile přidá první certifikát.

**Poznámka:** Z bezpečnostních důvodů neumožňuje toto rozhraní API vložit certifikát vydavatele certifikátu (CA) do paměti certifikátů \*SIGNATUREVERIFICATION. Když přidáte certifikát CA do paměti certifikátů, systém předpokládá, že vydavatel certifikátu (CA) je ověřeným zdrojem certifikátů. Následkem toho systém zachází s certifikátem vydaným CA, jako kdyby byl vydán ověřeným zdrojem. Z tohoto důvodu nemůžete používat rozhraní API k vytvoření programu uživatelského vstupu, který by vložil certifikát CA do paměti certifikátů. K přidání certifikátu CA do paměti certifikátů musíte použít produkt DCM (Digital Certificate Manager), abyste zajistili, že někdo musí speciálně a ručně řídit, kterým vydavatelům certifikátu (CA) systém důvěřuje. Tímto způsobem předejete možným případům, kdy systém mohl importovat certifikáty ze zdrojů, které administrátor vědomě nezadal jako důvěryhodné.


Pokud chcete komukoliv zabránit v používání tohoto rozhraní API k přidávání ověřovacího certifikátu do vaší paměti certifikátů \*SIGNATUREVERIFICATION bez vašeho vědomí, měli byste uvažovat o zablokování daného rozhraní API ve vašem systému. Tento úkon můžete provést pomocí nástrojů SST (system service tools), které zamítnou změny v systémových hodnotách týkajících se zabezpečení. .



Dříve byly informace o schopnostech serveru iSeries v oblasti podepisování objektů a ověření podpisu dostupné jako součást témata Digital Certificate Management v rámci aplikace Information Center. Nyní existují další metody, které můžete použít k podepsání objektů a ověření podpisů. Následkem toho je k dispozici toto nové téma v rámci aplikace Information Center, které zjednodušuje použití schopností podepisování objektů a ověřování podpisu, protože nabízí informace o použití těchto schopností z jednoho centrálního místa. Téma také nabízí rozšířené a podrobnější informace, jako např. scénáře, které vám pomohou určit, kdy a jak používat uvedené schopnosti obohacující vaši strategii zabezpečení ochrany dat.


Nové nebo zdokonalené informace o tomto tématu zahrnují:

- Scénáře, které vám mohou pomoci při stanovení nejlepšího způsobu využití schopností podepisování objektů a ověření podpisu pro doplnění vaší strategie zabezpečení ochrany dat.
- Nové sekce, které popisují příkazy a systémové hodnoty, které můžete používat ke správě podepsaných objektů ve vašem systému.
- Nové sekce, které popisují plánovací a další koncepční informace pro použití podepisování objektů a ověřování podpisů.

Další informace o tom, co je nového nebo co bylo změněno v této verzi produktu, viz dokument Sdělení uživatelům .

---

## Tisk tohoto tématu

Pokud chcete soubor typu PDF prohlížet nebo stáhnout, vyberte téma Podepisování objektů a ověření podpisu  (cca 350 KB nebo 44 stran).

Pokud chcete soubor ve formátu PDF uložit na svou pracovní stanici za účelem prohlížení nebo tisku:

1. Otevřete soubor typu PDF pomocí svého prohlížeče (klepněte na výše uvedený odkaz).
2. V menu vašeho prohlížeče klepněte na **Soubor**.
3. Klepněte na **Uložit jako...**
4. Vyhledejte adresář, do něhož chcete soubor typu PDF uložit.
5. Klepněte na **Uložit**.

Pokud ke stažení nebo prohlížení souboru ve formátu PDF potřebujete program Adobe Acrobat Reader, můžete si jeho kopii tohoto produktu stáhnout z domovské stránky společnosti Adobe

([www.adobe.com/prodindex/acrobat/readstep.html](http://www.adobe.com/prodindex/acrobat/readstep.html)) .

---

## Scénáře podepisování objektů

Váš server iSeries nabízí různé metody pro podepisování objektů a ověřování podpisů na objektech. To, který způsob podepisování objektů zvolíte a jak budete pracovat s podepsanými objekty, záleží na vašich obchodních potřebách a požadavcích v oblasti zabezpečení ochrany dat. V některých případech můžete potřebovat pouze ověřit podpisy objektů ve vašem systému, abyste se přesvědčili, že integrita objektu nebyla dotčena. V jiných případech se můžete rozhodnout podepisovat objekty, které distribuujete ostatním. Podepisování objektů umožňuje ostatním identifikovat původ objektů a zkontrolovat integritu objektů.

To, kterou metodu zvolíte, záleží na mnoha faktorech. Scénáře nabízené v tomto tématu popisují několik nejběžnějších cílů pro implementaci podepisování objektů a ověření podpisu v rámci typického podnikového prostředí. Každý scénář také popisuje všechny nezbytné předpoklady a úlohy, které je nezbytné provést při implementaci daného scénáře. Prostudování těchto scénářů vám pomůže určit takový způsob použití funkce podepisování objektů iSeries, který bude nejlépe odpovídat vašim obchodním potřebám i požadavkům v oblasti zabezpečení ochrany dat.

### **Scénář: Použití produktu DCM (Digital Certificate Manager) k podepisování objektů a ověřování podpisů**

Tento scénář popisuje společnost, která chce podepisovat nechráněné objekty aplikací na svém veřejném webovém serveru. Tato společnost chce mít možnost snadněji určit, kdy byly učiněny neoprávněné změny na těchto objektech. Na základě obchodních potřeb společnosti a na základě cílů v oblasti zabezpečení ochrany dat tento scénář popisuje, jak je možné používat produkt DCM (Digital Certificate Manager) jako primární způsob podepisování objektů a ověřování podpisů objektů.

### **Scénář: Použití rozhraní API k podepisování objektů a ověřování podpisů**

Tento scénář popisuje společnost zabývající se vývojem aplikací, která chce programově podepisovat aplikace, jež prodává. Chtějí mít možnost ujistit své zákazníky, že aplikace pochází z jejich společnosti a poskytnout jim prostředek pro detekci neautorizovaných změn v aplikacích během jejich instalace. Na základě obchodních potřeb společnosti a na základě cílů v oblasti zabezpečení ochrany dat tento scénář popisuje, jak je možné použít rozhraní Sign Object API a rozhraní Add Verifier API k podepisování objektů a umožnění ověření podpisu.

### **Scénář: Použití Centrální správy k podepisování objektů**

Tento scénář popisuje společnost, která chce podepisovat objekty, jež soustřeďuje do programových balíčků a distribuuje na více serverů iSeries. Na základě obchodních potřeb společnosti a na základě cílů v oblasti zabezpečení ochrany dat tento scénář popisuje, jak je možné použít funkci Centrální správa, která je komponentou produktu iSeries Navigator, k vytvoření programových balíčků a k podepsání objektů, jež budou distribuovány na jiné servery iSeries.

## **Scénář: Použití produktu DCM (Digital Certificate Manager) k podepisování objektů a ověřování podpisů**

### **Situace**

Jako administrátor systému iSeries společnosti MyCo., Inc. jste zodpovědný za správu dvou firemních serverů iSeries. Jeden z těchto serverů iSeries pracuje jako veřejný webový server vaší společnosti. Interní provozní server iSeries používáte k vytváření obsahu tohoto veřejného webového serveru a po otestování vytvořených souborů a objektů typu program je přenášíte na veřejný webový server.

Firemní veřejný webový server nabízí webové stránky s obecnými informacemi o společnosti. Webové stránky také poskytují různé formuláře, které zákazníci vyplňují, aby zaregistrovali produkty, vyžádali si informace o produktech, oznámění o aktualizaci produktů, lokalitách distribuce produktů apod. Máte obavy o zranitelnost programů cgi-bin, které jsou součástí těchto formulářů. Víte, že je možné je pozměnit. Proto chcete mít možnost kontrolovat integritu těchto objektů typu program a detekovat, kdy u nich byly provedeny neoprávněné změny. Z tohoto důvodu jste se rozhodli digitálně podepsat uvedené objekty, abyste splnili tento cíl zabezpečení vašich dat.

Pečlivě jste prozkoumali schopnosti podepisování objektu systému OS/400 a zjistili jste, že existuje několik způsobů, které můžete použít k podepisování objektů a ověření podpisů objektu. Jelikož jste odpovědní za správu malého počtu serverů iSeries a nezdá se vám, že budete muset často podepisovat objekty, rozhodli jste se používat produkt Digital Certificate Manager (DCM), pomocí kterého budete provádět uvedené úlohy. Dále jste se rozhodli vytvořit lokálního vydavatele certifikátu (CA) a používat soukromé certifikáty pro podepisování objektů. Používání soukromých certifikátů vydávaných lokálním CA pro podepisování objektů snižuje náklady na používání této bezpečnostní technologie, protože nemusíte zakoupit certifikát od známého veřejného vydavatele certifikátů (CA).

Tento příklad slouží jako užitečný úvod do procesu nastavení a používání podepisování objektů, pokud si přejete podepisovat objekty na malém počtu serverů iSeries.

### **Výhody scénáře**

Tento scénář má následující výhody:

- Podepisování objektů vám poskytuje prostředky ke kontrole integrity zranitelných objektů a možnost snadněji určit, zda byly objekty změněny poté, co byly podepsány. Tak můžete snížit výskyt a řešení některých problémů, které by se objevily v budoucnosti při pátrání po změnách v aplikacích a při jiných problémech systému.
- Používání grafického uživatelského rozhraní (GUI) produktu DCM vám umožňuje rychleji a snadněji podepisovat objekty a ověřovat podpisy objektů.
- Používání DCM k podepisování objektů a ověřování podpisů objektu snižuje dobu nutnou k pochopení a implementaci podepisování objektů do vaší strategie zabezpečení ochrany dat.
- Používání certifikátu, vydaného lokálním vydavatelem certifikátů (CA), k podepisování objektů snižuje náklady na implementaci.

## Cíle

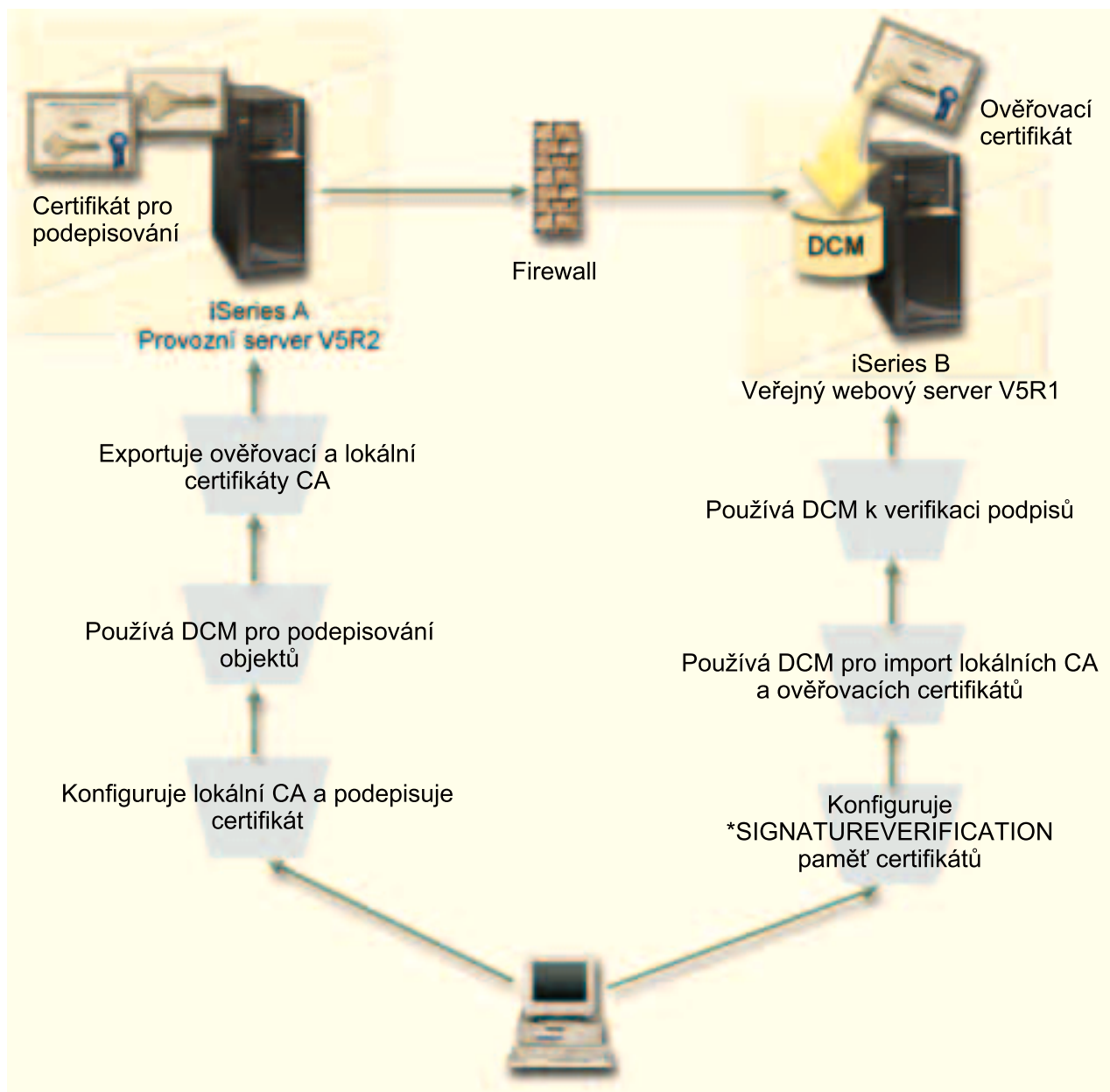
V tomto scénáři si přejete digitálně podepisovat zranitelné objekty, jako např. programy cgi-bin, které generují formuláře, na vašem firemním veřejném serveru iSeries. Jako systémový administrátor společnosti MyCo, Inc., chcete pro podepisování těchto objektů a ověření podpisů na objektech používat produkt DCM (Digital Certificate Manager).

Cíle tohoto scénáře jsou následující:

- Firemní aplikace a další zranitelné objekty na veřejném webovém serveru (iSeries B) musí být podepsány certifikátem od lokálního vydavatele certifikátů (CA), aby byly sníženy náklady na aplikaci podepisování.
- Systémoví administrátoři a další určené uživatelé musí být schopni snadno ověřit digitální podpisy na serverech iSeries, aby ověřili zdroj a pravost firemních podepsaných objektů. Aby výše uvedené požadavky byly splněny, musí mít každý server iSeries vlastní kopii firemního certifikátu pro ověření podpisu a certifikátu lokálního vydavatele certifikátů (CA) ve své paměti certifikátů \*SIGNATUREVERIFICATION.
- Ověřením podpisů na firemních aplikacích a jiných objektech mohou administrátoři iSeries a jiní určené uživatelé detekovat, zda byl obsah objektů od doby jejich podpisu změněn.
- Systémový administrátor musí používat DCM k podepisování objektů. Systémový administrátor a jiní určené uživatelé musí být schopni používat DCM k ověření podpisů objektů.

## Podrobnosti

Následující obrázek znázorňuje proces podepisování objektů a ověření podpisu, který bude v tomto scénáři implementován:



Obrázek zobrazuje následující body, které se vztahují k tomuto scénáři:

### Server iSeries A

- Server iSeries A má nainstalován a spuštěn operační systém OS/400 verze 5, vydání 2 (V5R2).
- Server iSeries A je interní provozní server společnosti a vývojová platforma pro veřejný webový server iSeries (iSeries B).
- Server iSeries A má nainstalovány komponenty Cryptographic Access Provider 128-bit for iSeries (5722-AC3).
- Server iSeries A má nainstalovány a nakonfigurovány produkty Digital Certificate Manager (OS/400, volba 34) a IBM HTTP Server (5722-DG1).
- Server iSeries A vystupuje jako lokální vydavatel certifikátů (CA) a certifikát pro podepisování objektů je uložen v tomto systému.

- Server iSeries A používá DCM k podepisování objektů a vystupuje jako primární systém podepisování objektů pro firemní veřejné aplikace a další objekty.
- Server iSeries A je nakonfigurován tak, aby umožňoval ověřování podpisů.

### Server iSeries B

- Server iSeries B má nainstalován a spuštěn operační systém OS/400 verze 5, vydání 1 (V5R1).
- Server iSeries B je firemní vnější veřejný webový server, který je umístěn za ochrannou bariérou společnosti.
- Server iSeries B má nainstalovanou komponentu Cryptographic Access Provider 128-bit (5722-AC3).
- Server iSeries B má nainstalovány a nakonfigurovány produkty Digital Certificate Manager (OS/400, volba 34) a IBM HTTP Server (5722-DG1).
- Server iSeries B nevystupuje jako lokální CA, ani nepodepisuje objekty.
- Server iSeries B je nakonfigurován tak, aby umožnil ověřování podpisu pomocí produktu DCM tak, že vytvoří paměť certifikátů \*SIGNATUREVERIFICATION a naimportuje potřebné certifikáty pro ověření a certifikáty lokálního vydavatele certifikátů (CA).
- Produkt DCM se používá k ověřování podpisů na objektech.

### Nezbytné podmínky a předpoklady

Nezbytné podmínky a předpoklady pro realizaci uvedeného scénáře jsou tyto:

1. Všechny servery iSeries splňují požadavky pro nainstalování a použití produktu DCM (Digital Certificate Manager).
2. Na žádném ze serverů iSeries dosud nikdo nekonfiguroval a nepoužíval produkt DCM.
3. Všechny servery iSeries mají nainstalovanou nejnovější licencovaného programu Cryptographic Access Provider 128-bit (5722-AC3).
4. Systémová hodnota QVfyOBJRST (Verify object signatures during restore) na všech serverech iSeries ve scénáři je předvoleně nastavena na hodnotu 3 a toto nastavení nebude změněno. Předvolené nastavení zajišťuje, aby server mohl ověřovat podpisy objektů, zatímco vy obnovujete podepsané objekty.
5. Systémový administrátor serveru iSeries A musí mít zvláštní oprávnění \*ALLOBJ, aby mohl podepisovat objekty, nebo jeho uživatelský profil musí mít oprávnění k aplikaci pro podepisování objektů.
6. Systémový administrátor nebo jiný uživatel, který vytváří paměť certifikátů v DCM, musí mít zvláštní oprávnění \*SECADM a \*ALLOBJ.
7. Systémový administrátor anebo jiný uživatelé serverů iSeries musí mít zvláštní oprávnění \*AUDIT, aby byli schopni ověřovat podpisy objektů.

### Kroky scénáře

Při realizaci tohoto scénáře musíte provést dvě skupiny úloh: V první skupině úloh budete konfigurovat server iSeries A tak, aby vystupoval jako lokální vydavatel certifikátů (CA) a aby byl schopen podepisovat a ověřovat podpisy objektů. Ve druhé skupině úloh budete konfigurovat server iSeries B tak, aby byl schopen ověřovat podpisy objektů, které vytvoří server iSeries A.

### Skupina úloh týkající se serveru iSeries A

Musíte dokončit každou z níže uvedených úloh na serveru iSeries A, abyste vytvořili soukromého lokálního vydavatele certifikátů (CA) a abyste byly schopni podepisovat objekty a ověřovat podpisy objektu způsobem, který je popsán ve scénáři:

1. Dokončete všechny kroky týkající se nezbytných předpokladů, vztahujících se k instalaci a konfiguraci potřebných produktů iSeries.

2. Pomocí produktu DCM vytvořte lokálního vydavatele certifikátů (CA), který vydá certifikát pro podepisování objektů.
3. Pomocí produktu DCM vytvořte definici aplikace.
4. Pomocí produktu DCM přiřaďte certifikát k definici aplikace pro podepisování objektů.
5. Pomocí produktu DCM podepište objekty programu cgi-bin.
6. Pomocí produktu DCM vyexportujte certifikáty, které musí používat další systémy k ověřování podpisů objektů. Musíte vyexportovat jak kopii certifikátu lokálního CA, tak i kopii certifikátu pro podepisování objektů do souboru. Obě kopie tvoří certifikát pro ověření podpisu.
7. Přeneste soubory s certifikáty na firemní veřejný server iSeries (iSeries B), takže vy i ostatní uživatelé můžete ověřovat podpisy, které vytvoří server iSeries A.

### Skupina úloh týkající se serveru iSeries B

Pokud máte v úmyslu obnovovat podepsané objekty, které v tomto scénáři přenesete na veřejný webový server (iSeries B), měli byste před vlastním přenosem podepsaných objektů provést na serveru iSeries B níže uvedené konfigurační úlohy týkající se ověřování podpisů. Konfiguraci ověřování podpisů musíte dokončit, abyste mohli úspěšně ověřovat podpisy během obnovování podepsaných objektů na veřejném webovém serveru.

Na serveru iSeries B je třeba provést níže uvedené úlohy, abyste byli schopni ověřovat podpisy na objektech způsobem, který je popsán ve scénáři:

8. Pomocí produktu DCM vytvořte paměť certifikátů \*SIGNATUREVERIFICATION.
9. Pomocí produktu DCM naimportujte certifikát lokálního vydavatele certifikátů (CA) a certifikát pro ověření podpisu.
10. Pomocí produktu DCM ověřte podpisy na přenesených objektech.

### Podrobnosti konfigurace

Chcete-li nakonfigurovat a používat produkt DCM k podepisování objektů způsobem, který je popsán v tomto scénáři, proveďte následující kroky:

#### Krok 1: Dokončete všechny kroky týkající se nezbytných předpokladů

K tomu, abyste mohli začít s konfiguračními úlohami při realizaci tohoto scénáře, musíte nejprve splnit všechny nezbytné předpoklady týkající se instalace a konfigurace potřebných produktů na serveru iSeries.

#### Krok 2: Vytvořte lokálního vydavatele certifikátů (CA), který bude vydávat soukromý certifikát pro podepisování objektů

Pokud chcete použít produkt DCM k vytvoření lokálního vydavatele certifikátů (CA), budete muset vyplnit řadu formulářů. Tyto formuláře vás provedou procesem vytvoření lokálního CA a dalšími úlohami potřebnými k zahájení používání digitálních certifikátů pro SSL (Secure Sockets Layer), podepisování objektů a ověřování podpisů. I když v tomto scénáři nemusíte nakonfigurovat certifikáty pro SSL, musíte vyplnit všechny formuláře uvedené v úloze, abyste mohli nakonfigurovat systém pro podepisování objektů.

Chcete-li pomocí produktu DCM vytvořit a provozovat lokálního CA, postupujte následovně:

1. Spusťte produkt DCM.
2. V navigačním okně produktu DCM vyberte volbu **Vytvoření vydavatele certifikátů (CA)**. Zobrazí se sada formulářů.

**Poznámka:** Jestliže si nejste jisti, jak vyplnit určitý formulář v této řízené úloze, vyberte tlačítko s otazníkem (?) v horní části stránky, čímž se dostanete do online nápovědy.

3. Vyplňte všechny formuláře pro tuto řízenou úlohu. Během provádění této úlohy musíte:

- a. Poskytnout identifikační informace pro lokálního CA.
- b. Nainstalovat certifikát lokálního CA do vašeho prohlížeče, aby váš software mohl rozpoznat lokálního CA a potvrzovat certifikáty, které lokální CA vydá.
- c. Zadat strategická data pro vašeho lokálního CA.
- d. Pomocí nového lokálního CA vydat serverový nebo klientský certifikát, který budou vaše aplikace používat pro připojení SSL.

**Poznámka:** Přestože tento scénář nepoužívá tento typ certifikátu, musíte jej vytvořit, abyste mohli pomocí lokálního CA vydávat certifikát pro podepisování objektů, který potřebujete. Pokud zrušíte úlohu, aniž byste vytvořili tento certifikát, musíte samostatně vytvořit váš certifikát pro podepisování objektů a paměť certifikátů \*OBJECTSIGNING, ve které je certifikát pro podepisování objektů uložen.

- e. Vybrat aplikace, které mohou použít serverový nebo klientský certifikát pro připojení SSL.

**Poznámka:** Pro účely tohoto scénáře nemusíte zvolit žádnou aplikaci a klepnutím na tlačítko **Pokračovat** zobrazíte další formulář.

- f. Pomocí nového lokálního CA vydat certifikát pro podepisování objektů, který budou používat aplikace k digitálnímu podepisování objektů. Tato podúloha vytvoří paměť certifikátů \*OBJECTSIGNING. Tuto paměť certifikátů budete používat při správě certifikátů pro podepisování objektů.
- g. Vybrat aplikace, které by měly důvěřovat vašemu lokálnímu CA.

**Poznámka:** Pro účely tohoto scénáře nemusíte zvolit žádnou aplikaci a klepnutím na tlačítko **Pokračovat** dokončíte úlohu.

Teď, když jste vytvořili lokálního vydavatele certifikátů (CA) a certifikát pro podepisování objektů, musíte nadefinovat aplikaci pro podepisování objektů, která bude uvedený certifikát pro podepisování objektů používat.

### Krok 3: Vytvořte aplikaci pro podepisování objektů

Když jste vytvořili váš certifikát pro podepisování objektů, musíte pomocí produktu DCM nadefinovat aplikaci pro podepisování objektů, která uvedený certifikát bude používat k podepisování objektů. Definice aplikace se nemusí odkazovat na skutečnou aplikaci. Definice aplikace, kterou vytvoříte, by měla místo toho popisovat typ nebo skupinu objektů, které hodláte podepisovat. Definici potřebujete, abyste mohli přiřadit ID aplikace k certifikátu a tak aktivovali proces podepisování.

Chcete-li pomocí produktu DCM vytvořit aplikaci pro podepisování objektů, postupujte takto:

1. V navigačním okně klepněte na **Výběr paměti certifikátů** a vyberte **\*OBJECTSIGNING** jako paměť certifikátů, kterou chcete otevřít.
2. Když se zobrazí stránka Paměť certifikátů a heslo, zadejte heslo, které jste pro danou paměť certifikátů uvedli, když jste ji tvořili, a klepněte na **Pokračovat**.
3. V navigačním okně vyberte volbu **Správa aplikací**. Zobrazí se seznam úloh.
4. Vyberte ze seznamu úloh volbu **Přidat aplikaci**. Zobrazí se formulář pro definici aplikace.
5. Vyplňte formulář a klepněte na **Přidat**.

Nyní musíte přiřadit váš certifikát pro podepisování objektů k aplikaci, kterou jste vytvořili.

### Krok 4: Přiřaďte certifikát k definici aplikace pro podepisování objektů

Chcete-li přiřadit certifikát k aplikaci pro podepisování objektů, proveďte následující kroky:

1. V navigačním okně DCM vyberte **Správa certifikátů**. Zobrazí se seznam úloh.

2. Ze seznamu úloh vyberte volbu **Přiřadit certifikát**. Zobrazí se seznam certifikátů v aktuální paměti certifikátů.
3. Ze seznamu vyberte příslušný certifikát a klepněte na **Přiřadit k aplikacím**. Zobrazí se seznam definic aplikací pro aktuální paměť certifikátů.
4. Vyberte ze seznamu jednu nebo více aplikací a klepněte na **Pokračovat**. Zobrazí se stránka, která buď potvrdí přiřazení certifikátu, nebo v případě problémů informuje o chybách.

Až dokončíte tuto úlohu, budete připraveni prostřednictvím DCM podepisovat objekty typu program, které se budou používat na firemním veřejném webovém serveru (iSeries B).

#### Krok 5: Podepište objekty typu program

Chcete-li prostřednictvím DCM podepisovat objekty typu program, které se budou používat na firemním veřejném webovém serveru (iSeries B), proveďte následující kroky:

1. V navigačním okně klepněte na **Výběr paměti certifikátů** a vyberte **\*OBJECTSIGNING** jako paměť certifikátů, kterou chcete otevřít.
2. Zadejte heslo pro paměť certifikátů **\*OBJECTSIGNING** a klepněte na **Pokračovat**.
3. Když se obnoví navigační okno, vyberte **Správa podepisovatelných objektů**. Zobrazí se seznam úloh.
4. Vyberte ze seznamu úlohu **Podepsat objekt** a zobrazí se seznam definic aplikací, které můžete použít pro podepisování objektů.
5. Vyberte aplikaci, kterou jste nedefinovali v předešlém kroku a klepněte na **Podepsat objekt**. Zobrazí se formulář, který vám umožní zadat umístění objektů, které si přejete podepsat.
6. Do nabídnutého pole zadejte úplnou cestu a jméno souboru objektu nebo adresáře objektů, které chcete podepsat, a klepněte na **Pokračovat**. Nebo zadejte umístění adresáře, klepněte na **Procházet**. Zobrazí se obsah adresáře, abyste mohli vybrat objekty pro podepsání.

**Poznámka:** Jméno objektu musíte začít úvodním lomítkem, jinak by se mohla vyskytnout chyba. Pro popis části adresáře, kterou chcete podepsat, můžete také použít určité zástupné znaky. Tyto zástupné znaky jsou hvězdička (\*), která zastupuje *libovolný počet znaků* a otazník (?), který zastupuje *libovolný jednotlivý znak*. Pokud např. chcete podepsat všechny objekty v určitém adresáři, můžete zadat `/mydirectory/*`; nebo když chcete podepsat všechny programy v určité knihovně, můžete zadat `/QSYS.LIB/QGPL.LIB/*.PGM`. Tyto zástupné znaky můžete používat pouze v poslední části jména cesty; zadání např. `/mydirectory*/filename` by mělo za následek chybovou zprávu. Pokud chcete použít funkci **Procházet**, abyste viděli seznam obsahu knihovny nebo adresáře, měli byste zadat zástupný znak jako součást jména cesty předtím, než klepnete na **Procházet**.

7. Vyberte volbu zpracování, kterou chcete použít k podepsání vybraného objektu nebo objektů, a klepněte na **Pokračovat**.

**Poznámka:** Pokud se rozhodnete čekat na výsledky úlohy, zobrazí se soubor s výsledky přímo ve vašem prohlížeči. Výsledky pro aktuální úlohu jsou připojeny ke konci souboru s výsledky. Soubor tudíž kromě výsledků aktuální úlohy může obsahovat výsledky z kterýchkoliv předchozích úloh. Pomocí pole data v souboru můžete určit, které řádky souboru se týkají aktuální úlohy. Pole data je ve formátu RRRRMMDD. První pole v souboru může být buď ID zprávy (pokud v průběhu zpracování objektu došlo k chybě), nebo pole data (udává datum zpracování úlohy).

8. Uveďte úplnou cestu a jméno souboru, do kterého se mají uložit výsledky úlohy podepsání objektu, a klepněte na **Pokračovat**. Anebo zadejte umístění adresáře, klepněte na **Procházet** a zobrazí se vám obsah adresáře, abyste mohli vybrat soubor pro uložení výsledků úlohy. Zobrazí se zpráva, která oznamuje, že úloha pro podepsání objektu byla spuštěna. Výsledky úlohy si můžete prohlédnout v úloze **QOBJSGNBAT** v protokolu úlohy.



Chcete-li zajistit, abyste vy a další určené uživatelé mohli ověřovat podpisy, musíte vyexportovat nezbytné certifikáty do souboru a přenést soubor certifikátů na server iSeries B. Musíte také dokončit všechny úlohy konfigurace ověřování podpisů na serveru iSeries B, než přenesete podepsané objekty typu program na server iSeries B. Konfigurace ověřování podpisů musí být dokončena, abyste mohli úspěšně ověřovat podpisy během obnovy podepsaných objektů na serveru iSeries B.

#### **Krok 6: Vyexportujte certifikáty, abyste povolili ověřování podpisů na serveru iSeries B**

Použití metody podepisování objektů k zajištění integrity obsahu předpokládá, že vy a další určené uživatelé máte prostředky k ověřování pravosti podpisu. K tomu, abyste ověřili podpisy objektů ve stejném systému, který tyto objekty podepisuje (iSeries A), musíte pomocí produktu DCM vytvořit paměť certifikátů \*SIGNATUREVERIFICATION. Tato paměť certifikátů musí obsahovat jak kopii certifikátu pro podepisování objektu, tak i kopii certifikátu CA pro vydavatele certifikátu, který vydal certifikát pro podepisování.

Chcete-li umožnit i jiným, aby mohli ověřovat podpisy, musíte také jim poskytnout kopii certifikátu, který podepisuje objekty. Pokud používáte lokálního vydavatele certifikátů (CA) k vydávání certifikátů, musíte těmto určeným uživatelům poskytnout také kopii certifikátu lokálního CA.

Jestliže si přejete pomocí produktu DCM ověřovat podpisy ve stejném systému, který podepsal objekty (server iSeries A v tomto scénáři), proveďte následující kroky:

1. V navigačním okně vyberte volbu **Vytvoření nové paměti certifikátů** a zvolte \***SIGNATUREVERIFICATION** jako paměť certifikátů, která se má vytvořit.
2. Klepnutím na **Ano** zkopírujete existující certifikáty pro podepisování objektů do nové paměti certifikátů jako certifikáty pro ověřování podpisů.
3. Uveďte heslo pro novou paměť certifikátů a klepněte na **Pokračovat**, abyste vytvořili paměť certifikátů. Nyní můžete prostřednictvím DCM ověřovat podpisy objektů ve stejném systému, který používáte pro podepisování objektů.

Pokud chcete pomocí produktu DCM vyexportovat kopii certifikátu lokálního CA a kopii certifikátu pro podepisování objektů jako certifikát pro ověřování podpisů, abyste mohli ověřovat podpisy objektů na jiných systémech (iSeries B), proveďte následující kroky:

1. V navigačním okně vyberte volbu **Správa certifikátů** a pak úlohu **Export certifikátu**.
2. Vyberte **Vydavatel certifikátu (CA)** a klepněte na **Pokračovat**. Zobrazí se seznam certifikátů CA, které můžete vyexportovat.
3. Vyberte ze seznamu certifikát lokálního CA, kterého jste vytvořili dříve, a klepněte na **Export**.
4. Zadejte **Soubor** jako vaše místo určení exportu a klepněte na **Pokračovat**.
5. Zadejte úplnou cestu a jméno souboru exportovaného certifikátu lokálního CA a klepněte na **Pokračovat**, abyste vyexportovali certifikát.
6. Klepnutím na **OK** ukončíte stránku pro potvrzení exportu. Nyní můžete vyexportovat kopii certifikátu pro podepisování objektů.
7. Znovu vyberte úlohu **Export certifikátu**.
8. Vyberte **Podepisování objektů** a zobrazí se seznam certifikátů pro podepisování objektů, které můžete vyexportovat.
9. Vyberte ze seznamu odpovídající certifikát pro podepisování objektů a klepněte na **Export**.
10. Zvolte **Soubor jako certifikát pro ověřování podpisů** jako místo určení a klepněte na **Pokračovat**.
11. Zadejte úplnou cestu a jméno souboru exportovaného certifikátu pro ověřování podpisů a klepněte na **Pokračovat**, abyste vyexportovali certifikát.

Nyní můžete tyto soubory přenést na koncový systém iSeries, na kterém chcete ověřovat podpisy, jež jste vytvořili pomocí certifikátu.

#### **Krok 7: Přeneste soubory certifikátů na firemní veřejný server iSeries B**

Soubory certifikátů, které jste vytvořili na serveru iSeries A, musíte přenést na server iSeries B (v tomto scénáři jde o firemní veřejný webový server), abyste je mohli nakonfigurovat pro ověřování objektů, které jste podepsali. K přenesení souborů certifikátů můžete použít několik různých metod. K přenesení souborů můžete například použít FTP (File Transfer Protocol) nebo distribuci balíků programů v Centrální správě.

#### **Krok 8: Úlohy ověřování podpisů: Vytvořte paměť certifikátů \*SIGNATUREVERIFICATION**

Jestliže chcete ověřovat podpisy objektů na serveru iSeries B (firemní veřejný webový server), pak musí mít server iSeries B kopii odpovídajících certifikátů pro ověřování podpisů v paměti certifikátů \*SIGNATUREVERIFICATION. Jelikož jste se rozhodli k podepisování objektů používat certifikát, vydaný lokálním vydavatelem certifikátů (CA - certificate authority), paměť certifikátů musí také obsahovat kopii certifikátu lokálního CA.

Chcete-li vytvořit paměť certifikátů \*SIGNATUREVERIFICATION, postupujte následovně:

1. Spusťte produkt DCM.
2. V navigačním okně produktu DCM vyberte volbu **Vytvoření nové paměti certifikátů** a zvolte **\*SIGNATUREVERIFICATION** jako paměť certifikátů, která se má vytvořit.

**Poznámka:** Jestliže si nejste jisti, jak vyplnit určitý formulář v této řízené úloze, vyberte tlačítko s otázníkem (?) v horní části stránky, čímž se dostanete do online nápovědy.

3. Uveďte heslo pro novou paměť certifikátů a klepněte na **Pokračovat**, abyste vytvořili paměť certifikátů. Nyní můžete naimportovat certifikáty do paměti certifikátů a používat je k ověřování podpisů objektů.

#### **Krok 9: Úlohy ověřování podpisů: Naimportujte certifikáty**

Pokud chcete ověřovat podpis na objektu, paměť certifikátů \*SIGNATUREVERIFICATION musí obsahovat kopii certifikátu pro ověřování podpisů. Jestliže je certifikát pro podepisování soukromým certifikátem, paměť certifikátů musí také obsahovat kopii certifikátu lokálního vydavatele certifikátů (CA), který vydal certifikát pro podepisování. V tomto scénáři byly vyexportovány oba certifikáty do souboru a tento soubor byl přenesen na každý koncový systém iSeries.

Chcete-li naimportovat tyto certifikáty do paměti certifikátů \*SIGNATUREVERIFICATION, postupujte následovně:

1. V navigačním okně produktu DCM klepněte na **Výběr paměti certifikátů** a jako paměť certifikátů, kterou chcete otevřít, zvolte **\*SIGNATUREVERIFICATION**.
2. Když se zobrazí stránka Paměť certifikátů a heslo, zadejte heslo, které jste pro danou paměť certifikátů uvedli, když jste ji tvořili, a klepněte na **Pokračovat**.
3. Když se navigační okno obnoví, vyberte volbu **Správa certifikátů**. Zobrazí se seznam úloh.
4. Ze seznamu úloh vyberte úlohu **Import certifikátu**.
5. Jako typ certifikátu, který budete importovat, vyberte **Vydavatel certifikátu (CA)** a klepněte na **Pokračovat**.

**Poznámka:** Musíte nejprve naimportovat certifikát lokálního CA a teprve poté soukromý certifikát pro ověřování podpisů, jinak import certifikátu pro ověřování podpisů selže.

6. Uveďte úplnou cestu a jméno souboru s certifikátem CA a klepněte na **Pokračovat**. Zobrazí se zpráva, která buď potvrdí úspěšnost importu certifikátu, nebo poskytne informace o chybách v případě, že import selhal.
7. Znovu vyberte úlohu **Import certifikátu**.
8. Jako typ certifikátu, který se má naimportovat, vyberte volbu **Ověřování podpisů** a klepněte na **Pokračovat**.

- Uveďte úplnou cestu a jméno souboru s certifikátem pro ověřování podpisů a klepněte na **Pokračovat**. Zobrazí se zpráva, která buď potvrdí úspěšnost importu certifikátu nebo poskytne informace o chybách v případě, že import selhal.

Nyní můžete pomocí produktu DCM ověřovat na serveru iSeries B podpisy na objektech, které jste vytvořili s odpovídajícím certifikátem pro podepisování objektů na serveru iSeries A.

#### Krok 10: Úlohy ověřování podpisů: Ověřte podpisy na objektech typu program

Chcete-li pomocí produktu DCM ověřovat podpisy na přenesených objektech typu program, postupujte takto:

- V navigačním okně vyberte volbu **Výběr paměti certifikátů** a vyberte **\*SIGNATUREVERIFICATION** jako paměť certifikátů, kterou chcete otevřít.
- Zadejte heslo pro paměť certifikátů **\*SIGNATUREVERIFICATION** a klepněte na **Pokračovat**.
- Když se obnoví navigační okno, vyberte volbu **Správa podepisovatelných objektů**. Zobrazí se seznam úloh.
- Ze seznamu úloh vyberte úlohu **Ověření podpisu objektu**, abyste specifikovali umístění objektů, u kterých chcete ověřit podpis.
- Do nabídnutého pole zadejte úplnou cestu a jméno souboru objektu nebo adresáře objektů, u kterých chcete ověřit podpisy, a klepněte na **Pokračovat**. Nebo zadejte umístění adresáře, klepněte na **Procházet**. Zobrazí se obsah adresáře, abyste mohli vybrat objekty pro ověření podpisu.

**Poznámka:** Pro popis části adresáře, kterou chcete ověřit, můžete také použít určité zástupné znaky. Tyto zástupné znaky jsou hvězdička (\*), která zastupuje *libovolný počet znaků* a otazník (?), který zastupuje *libovolný jednotlivý znak*. Pokud např. chcete podepsat všechny objekty v určitém adresáři, můžete zadat `/mydirectory/*`; nebo když chcete podepsat všechny programy v určité knihovně, můžete zadat `/QSYS.LIB/QGPL.LIB/*.PGM`. Tyto zástupné znaky můžete používat pouze v poslední části jména cesty; zadání např. `/mydirectory*/filename` by mělo za následek chybovou zprávu. Pokud chcete použít funkci **Procházet**, abyste viděli seznam obsahu knihovny nebo adresáře, měli byste zadat zástupný znak jako součást jména cesty předtím, než klepnete na **Procházet**.

6. Vyberte volby zpracování, které chcete použít k ověřování podpisu na zvoleném objektu nebo objektech, a klepněte na **Pokračovat**.

**Poznámka:** Pokud vyberete volbu čekat na výsledky úlohy, zobrazí se soubor s výsledky přímo ve vašem prohlížeči. Výsledky pro aktuální úlohu jsou připojeny ke konci souboru s výsledky. Soubor tudíž kromě výsledků aktuální úlohy může obsahovat výsledky z kterýchkoliv předchozích úloh. Pomocí pole data v souboru můžete určit, které řádky souboru se týkají aktuální úlohy. Pole data je ve formátu RRRRMMDD. První pole v souboru může být buď ID zprávy (pokud v průběhu zpracování objektu došlo k chybě), nebo pole data (udává datum zpracování úlohy).

7. Uveďte úplnou cestu a jméno souboru, který se má použít pro uložení výsledků úlohy ověření podpisu, a klepněte na **Pokračovat**. Anebo zadejte umístění adresáře, klepněte na **Procházet** a zobrazí se vám obsah adresáře, abyste mohli vybrat soubor pro uložení výsledků úlohy. Zobrazí se zpráva, která oznamuje, že úloha pro ověření podpisů objektů byla spuštěna. Výsledky úlohy si můžete prohlédnout v úloze **QOBSJGNBAT**, v protokolu úlohy.

## Scénář: Použití rozhraní API k podepisování objektů a ověřování podpisů

### Situace

Vaše společnost (MyCo, Inc.) je iSeries obchodní partner, který vyvíjí aplikace pro zákazníky. Vy, jako vývojový pracovník softwaru společnosti, jste odpovědný za balení těchto aplikací pro distribuci k zákazníkům. V současné době používáte určité programy, které provádějí balení těchto aplikací. Zákazníci si mohou objednat kompaktní disk (CD-ROM) nebo si mohou aplikaci stáhnout z vaší webové stránky.

Snažíte se, abyste věděl o všech aktuálních novinkách v oboru, hlavně o novinkách v zabezpečení ochrany dat. Proto víte, že zákazníci mají oprávněné obavy o zdroj a obsah programů, které získávají nebo stahují z webové stránky. Stalo se, že si zákazníci mysleli, že získávají nebo stahují produkt z důvěryhodného zdroje, ale zjistilo se, že nešlo o skutečný zdroj produktu. Někdy toto nedorozumění vedlo k tomu, že zákazníci si nainstalovali jiný produkt, než který očekávali. Někdy se zjistilo, že nainstalovaný produkt je svévolný program nebo že nainstalovaný produkt byl změněn a poškodil systém zákazníka.

Ačkoli se tyto typy problémů u zákazníků se systémy iSeries běžně nevyskytují, chcete přesvědčit zákazníky, že aplikace, které od vás získají, jsou skutečně z vaší společnosti. Chcete také nabídnout zákazníkům možnost kontroly integrity těchto aplikací, aby si mohli sami zjistit, zda obdržené aplikace byly změněny, dříve, než je nainstalují.

Na základě vašeho zkoumání jste se rozhodli, že budete k dosažení vašich cílů v oblasti zabezpečení ochrany dat používat schopnosti podepisování objektů systému OS/400. Digitální podpisy na vašich aplikacích dovolí vašim zákazníkům ověřit, že vaše společnost je legitimním zdrojem aplikací, které obdrželi nebo stáhli. Jelikož v současné době balíte aplikace pomocí programu, rozhodli jste se, že použijete rozhraní API, abyste do vašeho aktuálního procesu balení snadno přidali podepisování objektů. Dále jste se rozhodli používat pro podepisování objektů veřejný certifikát, takže celý proces ověřování podpisů bude pro vaše zákazníky během instalace vašeho produktu transparentní.

Součástí balíku aplikací je i kopie digitálního certifikátu, který jste použili k podepsání objektů. Když zákazník obdrží balík aplikací, může pomocí veřejného klíče certifikátu ověřit podpis na aplikaci. Zákazník takto může identifikovat a ověřit zdroj aplikace, a zároveň si ověří, že obsah objektů aplikace nebyl od okamžiku podpisu změněn.

Tento příklad slouží jako užitečný úvod do procesu nastavení programového podepisování objektů pro aplikace, které vyvíjíte a balíte pro jiné uživatele.

## Výhody scénáře

Tento scénář má následující výhody:

- Použití rozhraní API k balení a podepisování objektů pomocí programu snižuje dobu, kterou musíte strávit nad implementací takového zabezpečení ochrany dat.
- Použití rozhraní API k podepisování objektů během jejich balení snižuje počet nezbytných kroků, které musíte provést při podepisování objektů, protože proces podepisování objektů je součástí procesu balení.
- Podepisování objektů vám poskytuje prostředky ke snadnější kontrole, zda objekty byly změněny po jejich podpisu. Tak můžete snížit výskyt a řešení některých problémů, které by se objevily v budoucnosti při pátrání po problémech v aplikacích u zákazníků.
- Používání certifikátu od známého veřejného vydavatele certifikátů (CA) při podepisování objektů vám dovoluje používat rozhraní Add Verifier API jako součást programu výstupního bodu ve vašem programu instalace produktu. Používání uvedeného rozhraní API vám umožňuje automaticky přidat veřejný certifikát, který jste použili k podepsání aplikace, do systému vašeho zákazníka. Tak zajistíte, aby ověřování podpisu bylo pro vašeho zákazníka transparentní.

## Cíle

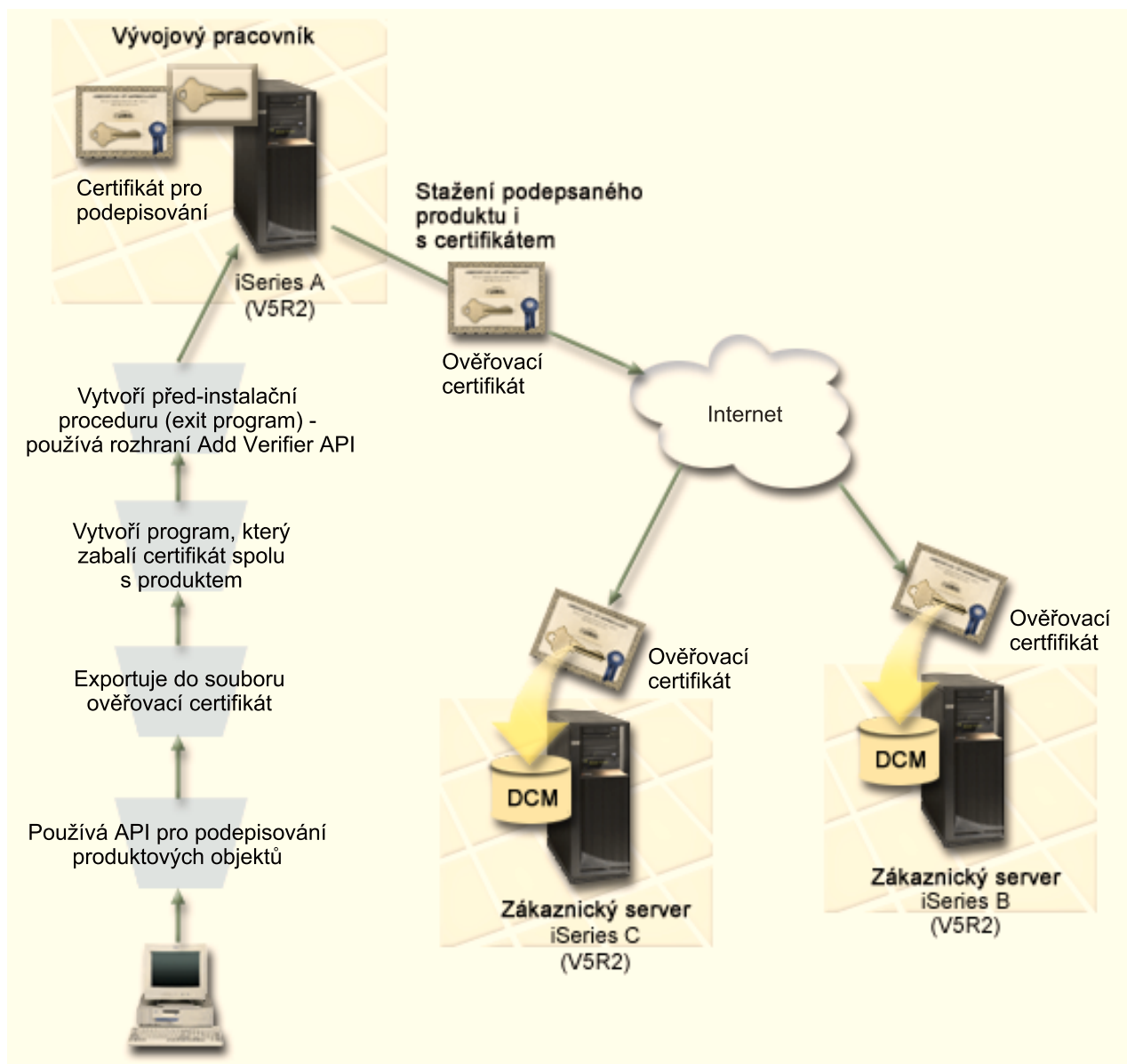
V tomto scénáři si společnost MyCo, Inc. přeje pomocí programu podepisovat aplikace, které balí a distribuuje svým zákazníkům. Jako vývojový pracovník výroby aplikací ve společnosti MyCo, Inc., v současné době balíte aplikace vaší společnosti pomocí programu, aby je bylo možné distribuovat zákazníkům. Proto chcete používat iSeries API k podepisování vašich aplikací a přimět systémy iSeries u zákazníka, aby ověřovaly podpis během instalace produktu.

Cíle tohoto scénáře jsou následující:

- Vývojový pracovník výroby aplikací musí být schopen podepisovat objekty pomocí rozhraní Sign Object API, které bude součástí stávajícího procesu balení aplikací pomocí programů.
- Aplikace společnosti musí být podepsány s veřejným certifikátem, aby bylo zajištěno, že proces ověřování podpisů bude pro zákazníky během procesu instalace aplikačního produktu transparentní.
- Společnost musí být schopna používat rozhraní iSeries API takovým způsobem, aby pomocí programu přidala požadovaný certifikát pro ověřování podpisů do paměti certifikátů \*SIGNATUREVERIFICATION na serveru iSeries zákazníka. Společnost musí být schopna vytvořit uvedenou paměť certifikátů na serveru iSeries zákazníka pomocí programu během procesu instalace produktu, pokud tato dosud neexistuje.
- Zákazníci musí být schopni po instalaci produktu snadno ověřit digitální podpisy na aplikacích společnosti. Zákazníci musí být schopni ověřit podpis, takže budou moci zjistit zdroj a pravost podepsaných aplikací a současně určit, zda v aplikacích byly provedeny změny po jejich podepsání.

#### **Podrobnosti**

Následující obrázek znázorňuje proces podepisování objektů a ověření podpisu, který bude implementován v tomto scénáři:



Obrázek zobrazuje následující body, které se vztahují k tomuto scénáři:

### Centrální systém (server iSeries A)

- Server iSeries A má nainstalován a spuštěn operační systém OS/400 verze 5, vydání 2 (V5R2).
- Server iSeries A má nainstalován a spuštěn program pro balení produktu, určený pro vývojáře aplikací.
- Server iSeries A má nainstalovány komponenty Cryptographic Access Provider 128-bit for iSeries (5722-AC3).
- Server iSeries A má nainstalovány a nakonfigurovány produkty Digital Certificate Manager (OS/400, volba 34) a IBM HTTP Server (5722-DG1).
- Server iSeries A je primárním systémem pro podepisování objektů pro aplikační produkty společnosti. Podepisování produktových objektů, určených k distribuci zákazníkům, je prováděno na serveru A pomocí těchto úloh:
  1. Rozhraní API se použije k podepisování aplikačních produktů společnosti.

2. Produkt DCM se použije k vyexportování certifikátu pro ověřování podpisu do souboru, takže zákazníci mohou ověřovat podepsané objekty.
3. Napiše se program, který přidá certifikát pro ověření podpisu do podepsaného aplikačního produktu.
4. Napiše se předinstalační program výstupního bodu produktu, který bude používat rozhraní Add Verifier API. Toto rozhraní API umožní, aby proces instalace produktu pomocí programu přidal certifikát pro ověřování podpisů do paměti certifikátů \*SIGNATUREVERIFICATION na zákaznickově serveru iSeries (iSeries B a C).

### Servery iSeries B a C zákazníka

- Server iSeries B má nainstalován a spuštěn operační systém OS/400 verze 5, vydání 2 (V5R2).
- Server iSeries C má nainstalován a spuštěn operační systém OS/400 verze 5, vydání 2 (V5R2).
- Servery iSeries B a C mají nainstalovány a nakonfigurovány produkty Digital Certificate Manager (volba 34) a IBM HTTP Server (5722–DG1).
- Servery iSeries B a C zakoupily a stáhly aplikaci z webové stránky společnosti vyvíjející aplikaci (která vlastní server iSeries A).
- Servery iSeries B a C obdržely kopii certifikátu pro ověřování podpisu společnosti MyCo během procesu instalace aplikace společnosti MyCo, který vytvořil paměť certifikátů \*SIGNATUREVERIFICATION na každém z těchto serverů iSeries zákazníka.

### Nezbytné podmínky a předpoklady

Nezbytné podmínky a předpoklady pro realizaci uvedeného scénáře jsou tyto:

1. Všechny servery iSeries splňují požadavky pro nainstalování a použití produktu DCM (Digital Certificate Manager).

**Poznámka:** Splnění nezbytných podmínek pro instalaci a použití produktu DCM je volitelným požadavkem pro zákazníky (servery iSeries B a C v tomto scénáři). Přestože rozhraní Add Verifier API vytvoří paměť certifikátů \*SIGNATUREVERIFICATION během procesu instalace produktu (pokud je to potřeba), vytvoří tuto paměť certifikátů s předvoleným heslem. Zákazníci musí použít produkt DCM ke změně předvoleného hesla, aby ochránili tuto paměť certifikátů před neoprávněnými přístupy.

2. Na žádném ze serverů iSeries dosud nikdo nekonfiguroval a nepoužíval produkt DCM.
3. Všechny servery iSeries mají nainstalovanou nejnovější úroveň licencovaného programu Cryptographic Access Provider 128-bit (5722-AC3).
4. Systémová hodnota QVFYOBJRST (Verify object signatures during restore) na všech serverech iSeries ve scénáři je předvoleně nastavena na hodnotu 3 a toto nastavení nebude změněno. Předvolené nastavení zajišťuje, aby server mohl ověřovat podpisy objektů, zatímco vy obnovujete podepsané objekty.
5. Systémový administrátor serveru iSeries A musí mít zvláštní oprávnění \*ALLOBJ, aby mohl podepisovat objekty, nebo jeho uživatelský profil musí mít oprávnění k aplikaci pro podepisování objektů.
6. Systémový administrátor nebo jiný uživatel (včetně programu), který vytváří paměť certifikátů v DCM, musí mít zvláštní oprávnění \*SECADM a \*ALLOBJ.
7. Systémoví administrátoři anebo jiní uživatelé dalších serverů iSeries musí mít zvláštní oprávnění \*AUDIT, aby byli schopni ověřovat podpisy objektů.

### Kroky scénáře

Musíte dokončit každou z níže uvedených úloh na serveru iSeries A, abyste byli schopni podepisovat objekty způsobem, který je popsán ve scénáři:

1. Dokončete všechny kroky týkající se nezbytných předpokladů, vztahující se k instalaci a konfiguraci potřebných produktů iSeries.

2. S použitím produktu DCM vytvořte paměť certifikátů, do které bude uložen certifikát pro podepisování objektů od známého veřejného vydavatele certifikátů (CA).
3. Pomocí produktu DCM vytvořte definici aplikace pro podepisování objektů.
4. Pomocí produktu DCM naimportujte podepsaný certifikát pro podepisování objektů a přiřaďte jej k vaší definici aplikace pro podepisování objektů.
5. Pomocí produktu DCM vyexportujte váš certifikát pro podepisování objektů jako certifikát pro ověřování podpisů, takže vaši zákazníci jej budou moci používat k ověřování podpisu na vašich objektech aplikace.
6. Přepište váš program pro balení aplikace tak, aby se soubor certifikátů pro ověřování podpisů stal součástí produktu a aby používal rozhraní Sign Object API k podepisování vaší aplikace během jejího balení pro účely distribuce k zákazníkům.
7. Vytvořte předinstalační program výstupního bodu, který bude používat rozhraní Add Verifier API a který se stane součástí vašeho procesu balení aplikace. Tento program výstupního bodu vám umožní během instalace produktu vytvořit paměť certifikátů \*SIGNATUREVERIFICATION a přidat požadovaný certifikát pro ověřování podpisů na zákaznickém serveru iSeries.
8. Přimějte zákazníky používat produkt DCM k vynulování předvoleného hesla paměti certifikátů \*SIGNATUREVERIFICATION na jejich serveru iSeries.

### Podrobnosti konfigurace

Chcete-li používat rozhraní API operačního systému OS/400 k podepisování objektů způsobem, který je popsán v tomto scénáři, proveďte následující kroky.

#### Krok 1: Dokončete všechny kroky týkající se nezbytných předpokladů

K tomu, abyste mohli začít s konfiguračními úlohami při realizaci tohoto scénáře, musíte nejprve splnit všechny nezbytné předpoklady týkající se instalace a konfigurace potřebných produktů na serveru iSeries.

#### Krok 2: Pomocí produktu DCM získáte certifikát od známého veřejného vydavatele certifikátů (CA)

Tento scénář předpokládá, že jste nikdy dříve nepoužívali produkt DCM (Digital Certificate Manager) k vytváření a správě certifikátů. Proto součástí procesu vytvoření vašeho certifikátu pro podepisování objektů musí být vytvoření paměti certifikátů \*OBJECTSIGNING. Uvedená paměť certifikátů, je-li vytvořena, obstarává úlohy, potřebné k vytvoření a správě certifikátů pro podepisování objektů. Chcete-li získat certifikát od známého veřejného vydavatele certifikátů (CA), vytvořte pomocí produktu DCM identifikační informace a pár klíčů veřejný-soukromý, určený pro certifikát, a předejte tyto informace vydavateli certifikátů, který vám předá váš certifikát.

Chcete-li vytvořit informace, které musíte předat známému veřejnému vydavateli certifikátů pro účely vydání certifikátu pro podepisování objektů, postupujte takto:

1. Spusťte produkt DCM.
2. V navigačním okně produktu DCM vyberte volbu **Vytvoření nové paměti certifikátů**. Tím spustíte řízenou úlohu, pomocí které vyplníte sadu formulářů. Pomocí těchto formulářů budete provedeni procesem vytvoření paměti certifikátů a certifikátu, který můžete používat pro podepisování objektů.

**Poznámka:** Jestliže si nejste jisti, jak vyplnit určitý formulář v této řízené úloze, vyberte tlačítko s otazníkem (?) v horní části stránky, čímž se dostanete do online nápovědy.

3. Vyberte **\*OBJECTSIGNING** jako paměť certifikátů, kterou chcete vytvořit, a klepněte na **Pokračovat**.
4. Vyberte **Ano**, abyste v rámci vytvoření paměti certifikátů \*OBJECTSIGNING vytvořili i certifikát a klepněte na **Pokračovat**.
5. Vyberte **VeriSign nebo jiný internetový vydavatel certifikátů (CA)** jako toho, kdo bude podepisovat nové certifikáty, a klepněte na **Pokračovat**, čímž se vám zobrazí formulář na vložení identifikačních informací pro nový certifikát.



6. Vyplňte formulář a klepnutím na **Pokračovat** se vám zobrazí potvrzující stránka. Tato potvrzující stránka zobrazuje data požadavku na certifikát, která musíte poskytnout veřejnému vydavateli certifikátu (CA), který bude váš certifikát vydávat. Data tohoto tzv. požadavku na podepisovací certifikát (Certificate Signing Request, CSR) zahrnují veřejný klíč a další informace, které jste uvedli pro nový certifikát.
7. Pečlivě zkopírujte a vložte data CSR do formuláře žádosti o certifikát nebo do zvláštního souboru, který veřejný CA požaduje při žádostech o certifikát. Musíte použít veškerá data CSR, včetně řádek Begin a End New Certificate Request. Jakmile tuto stránku opustíte, budou data ztracena a nebude možné je obnovit.
8. Pošlete formulář žádosti nebo soubor vydavateli CA, kterého jste si zvolili pro vydání a podepsání vašeho certifikátu.
9. Než budete moci pokračovat, musíte počkat, až vám CA vrátí podepsaný dokončený certifikát.

### Krok 3: Vytvořte definici aplikace pro podepisování objektů

Když jste odeslali váš požadavek na vydání certifikátu známému veřejnému vydavateli certifikátů, můžete pomocí produktu DCM nadefinovat aplikaci pro podepisování objektů, kterou budete používat k podepisování objektů. Definice aplikace se nemusí odkazovat na skutečnou aplikaci. Definice aplikace, kterou vytvoříte, by měla místo toho popisovat typ nebo skupinu objektů, které hodláte podepisovat. Definici potřebujete, abyste mohli přiřadit ID aplikace k certifikátu a tak aktivovali proces podepisování.

Chcete-li pomocí produktu DCM vytvořit definici aplikace pro podepisování objektů, postupujte takto:

1. V navigačním okně klepněte na **Výběr paměti certifikátů** a vyberte **\*OBJECTSIGNING** jako paměť certifikátů, kterou chcete otevřít.
2. Když se zobrazí stránka Paměť certifikátů a heslo, zadejte heslo, které jste pro danou paměť certifikátů uvedli, když jste ji tvořili, a klepněte na **Pokračovat**.
3. V navigačním okně vyberte volbu **Správa aplikací**. Zobrazí se seznam úloh.
4. Ze seznamu úloh vyberte úlohu **Přidat aplikaci**. Zobrazí se formulář pro definici aplikace.
5. Vyplňte formulář a klepněte na **Přidat**.

Jakmile obdržíte podepsaný certifikát od veřejného CA, můžete přiřadit certifikát k aplikaci, kterou jste vytvořili.

### Krok 4: Naimportujte podepsaný veřejný certifikát a přiřaďte jej k aplikaci pro podepisování objektů

Při importu vašeho certifikátu a jeho přiřazení k aplikaci, čímž aktivujete podepisování objektů, postupujte takto:

1. Spusťte produkt DCM.
2. V navigačním okně klepněte na **Výběr paměti certifikátů** a vyberte **\*OBJECTSIGNING** jako paměť certifikátů, kterou chcete otevřít.
3. Když se zobrazí stránka Paměť certifikátů a heslo, zadejte heslo, které jste pro danou paměť certifikátů uvedli, když jste ji tvořili, a klepněte na **Pokračovat**.
4. Když se navigační okno obnoví, vyberte volbu **Správa certifikátů**. Zobrazí se seznam úloh.
5. Ze seznamu úloh vyberte úlohu **Import certifikátu**, čímž zahájíte proces importu podepsaného certifikátu do paměti certifikátů.

**Poznámka:** Jestliže si nejste jisti, jak vyplnit určitý formulář v této řízené úloze, vyberte tlačítko s otazníkem (?) v horní části stránky, čímž se dostanete do online nápovědy.

6. Ze seznamu úloh **Správa certifikátů** vyberte úlohu **Přiřadit certifikát**. Zobrazí se seznam certifikátů v aktuální paměti certifikátů.
7. Ze seznamu vyberte příslušný certifikát a klepněte na **Přiřadit k aplikacím**. Zobrazí se seznam definic aplikací pro aktuální paměť certifikátů.

8. Vyberte ze seznamu vaši aplikaci a klepněte na **Pokračovat**. Zobrazí se stránka buď se zprávou potvrzující zvolené přiřazení, nebo s chybovou zprávou v případě nějakého problému.

Až dokončíte tuto úlohu, budete připraveni podepisovat aplikace a jiné objekty prostřednictvím rozhraní API operačního systému OS/400. Pokud však chcete zajistit, abyste vy i jiní určené uživatele mohli ověřovat podpisy, musíte vyexportovat nezbytné certifikáty do souboru a přenést je na všechny servery iSeries, na kterých se budou instalovat vaše podepsané aplikace. Zákazník se serverem iSeries pak musí být schopen používat certifikát pro účely ověření podpisu na vaší aplikaci během její instalace. Součástí vašeho instalačního programu aplikace může být rozhraní Add Verifier API, abyste jeho prostřednictvím provedli nezbytnou konfiguraci ověřování podpisů na straně vašeho zákazníka. Můžete například vytvořit předinstalační program výstupního bodu, který pomocí rozhraní Add Verifier API nakonfiguruje server iSeries u vašeho zákazníka.

#### **Krok 5: Vyexportujte certifikáty, abyste povolili ověřování podpisů na ostatních serverech iSeries**

Použití metody podepisování objektů vyžaduje, abyste vy a další určené uživatele měli prostředky k ověřování pravosti podpisu a možnost používat tyto prostředky k určení, zda na podepsaném objektu byly provedeny nějaké změny. K ověřování podpisů objektů ve stejném systému, který tyto objekty podepisuje, musíte pomocí produktu DCM vytvořit paměť certifikátů \*SIGNATUREVERIFICATION. Uvedená paměť certifikátů musí obsahovat kopii jak certifikátu pro podepisování objektů, tak i kopii certifikátu CA pro CA, který vydal certifikát pro podepisování.

Chcete-li umožnit i jiným, aby mohli ověřovat podpisy, musíte také jim poskytnout kopii certifikátu, který podepisuje objekty. Pokud používáte lokálního vydavatele certifikátů (CA) k vydávání certifikátů, musíte těmto určeným uživatelům poskytnout také kopii certifikátu lokálního CA.

Jestliže si přejete pomocí produktu DCM ověřovat podpisy ve stejném systému, který podepsal objekty (server iSeries A v tomto scénáři), proveďte následující kroky:

1. V navigačním okně vyberte volbu **Vytvoření nové paměti certifikátů** a zvolte **\*SIGNATUREVERIFICATION** jako paměť certifikátů, která se má vytvořit.
2. Klepnutím na **Ano** zkopírujete existující certifikáty pro podepisování objektů do nové paměti certifikátů jako certifikáty pro ověřování podpisů.
3. Uveďte heslo pro novou paměť certifikátů a klepněte na **Pokračovat**, abyste vytvořili paměť certifikátů. Nyní můžete prostřednictvím DCM ověřovat podpisy objektů ve stejném systému, který používáte pro podepisování objektů.

Pokud chcete pomocí produktu DCM vyexportovat kopii certifikátu pro podepisování objektů jako certifikát pro ověřování podpisů, abyste umožnili i jiným uživatelům ověřovat vaše podpisy na objektech, postupujte takto:

1. V navigačním okně vyberte volbu **Správa certifikátů** a pak úlohu **Export certifikátu**.
2. Vyberte volbu **Podepisování objektů**. Zobrazí se seznam certifikátů pro podepisování objektů, které můžete vyexportovat.
3. Vyberte ze seznamu odpovídající certifikát pro podepisování objektů a klepněte na **Export**.
4. Zvolte **Soubor jako certifikát pro ověřování podpisů** jako místo určení a klepněte na **Pokračovat**.
5. Zadejte úplnou cestu a jméno souboru exportovaného certifikátu pro ověřování podpisů a klepněte na **Pokračovat**, abyste vyexportovali certifikát.

Nyní můžete přidat tento soubor do instalačního balíku programů aplikace, který pro váš produkt vytváříte. Pokud použijete rozhraní Add Verifier API jako součást vašeho instalačního programu, můžete přidat tento certifikát do zákaznickovy paměti certifikátů \*SIGNATUREVERIFICATION. Rozhraní API vytvoří uvedenou paměť certifikátů, pokud dosud neexistuje. Váš instalační program pak může ověřit podpis na vašich objektech aplikace během jejich obnovení na serverech iSeries na straně zákazníka.

## Krok 6: Aktualizujte váš program pro balení aplikace tak, aby používal rozhraní API systému iSeries k podepisování vašich aplikací

Nyní, když máte soubor s certifikátem pro ověřování podpisů přidán do vašeho aplikačního balíku programů, můžete pomocí rozhraní Sign Object API psát nebo upravovat stávající aplikaci tak, aby podepisovala vaše knihovny produktu během jejich balení pro účely distribuce k zákazníkům.

Proto, abychom vám pomohli lépe porozumět, jak použít rozhraní Sign Object API v programu pro balení aplikace, prohlédněte si níže uvedený příklad kódu. Tento příklad části kódu, psaného v jazyce C, není úplným programem pro podepisování a balení. Jde spíše o příklad části takového programu, která volá rozhraní Sign Object API. Jestliže se rozhodnete použít tento příklad programu, změňte jej tak, aby odpovídal vašim specifickým požadavkům. Z bezpečnostních důvodů IBM doporučuje, abyste příklad programu implementovali vlastním způsobem, než abyste použili zde uvedené předvolené hodnoty.

**Poznámka:** IBM vám uděluje nevýhradní autorskou licenci k používání všech příkladů programových kódů, ze kterých můžete vygenerovat podobné funkce, které budou odpovídat vašim vlastním specifickým požadavkům. Všechny příklady kódů poskytuje IBM pouze pro účely ilustrace. Uvedené příklady nebyly důkladně testovány za všech podmínek. Z tohoto důvodu IBM nemůže zaručit ani odvodit spolehlivost, funkčnost nebo provoz těchto programů. Všechny programy zde obsažené jsou vám nabízeny "tak jak jsou", bez záruky jakéhokoliv druhu. Výslovně se zříkáme odvozených záruk neporušení, prodejnosti a vhodnosti pro určitý účel.

Změňte tento úryvek kódu tak, aby odpovídal vašim požadavkům a potřebám na použití rozhraní Sign Object API v programu pro balení vašeho aplikačního produktu. Do tohoto programu musíte předat dva parametry: jméno knihovny, která se má podepsat, a jméno ID aplikace pro podepisování podpisů. ID aplikace rozlišuje velká a malá písmena, jméno knihovny nikoliv. Program, který napíšete, může volat tento úryvek programu vícekrát, pokud součástí produktu je více knihoven, které je potřeba podepsat.

```
/* ----- */
/* */
/* 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;
}

```

**Krok 7: Vytvořte předinstalační program výstupního bodu, který bude používat rozhraní Add Verifier API**

Nyní když máte programový proces pro podepisování vaší aplikace, můžete jako součást vašeho instalačního programu používat rozhraní Add Verifier API k vytvoření finálního produktu pro distribuci. Můžete například použít rozhraní Add Verifier API jako součást předinstalačního programu výstupního bodu, abyste zajistili, že se certifikát přidá do paměti certifikátů dříve, než se obnoví podepsané objekty aplikace. Váš instalační program pak může ověřit podpis na vašich objektech aplikace během jejich obnovení na serverech iSeries na straně zákazníka.

**Poznámka:** Z bezpečnostních důvodů toto rozhraní API neumožňuje vložit certifikát vydavatele certifikátu (CA) do paměti certifikátů \*SIGNATUREVERIFICATION. Když přidáte certifikát CA do paměti certifikátů, systém předpokládá, že CA je ověřený zdroj certifikátů. Následkem toho systém zachází s certifikátem vydaným CA, jako kdyby byl vydán ověřeným zdrojem. Z tohoto důvodu nemůžete používat rozhraní API k vytvoření programu uživatelského vstupu, který by vložil certifikát CA do paměti certifikátů. K přidání certifikátu CA do paměti certifikátů musíte použít produkt DCM (Digital Certificate Manager), abyste zajistili, že někdo musí speciálně a ručně řídit, kterým vydavatelům certifikátu (CA) systém důvěřuje. Tímto způsobem předejdete možným případům, kdy systém mohl importovat certifikáty ze zdrojů, které administrátor vědomě nezadal jako důvěryhodné.

Pokud chcete komukoliv zabránit v používání tohoto rozhraní API k přidávání ověřovacího certifikátu do vaší paměti certifikátů \*SIGNATUREVERIFICATION bez vašeho vědomí, měli byste uvažovat o zablokování daného rozhraní API ve vašem systému. Tento úkon můžete provést pomocí nástrojů SST (system service tools), které zamítnou změny v systémových hodnotách týkajících se zabezpečení. .

Proto, abyste lépe porozuměli, jak používat rozhraní Add Verifier API v instalačním programu aplikace, prohlédněte si následující příklad kódu předinstalačního programu výstupního bodu. Tento příklad části kódu, psaného v jazyce C, není úplným předinstalačním programem výstupního bodu. Jde spíše o příklad části takového programu, která volá rozhraní Add Verifier API. Jestliže se rozhodnete použít tento příklad programu, změňte jej tak, aby odpovídal vašim specifickým požadavkům. Z bezpečnostních důvodů IBM doporučuje, abyste příklad programu spíše implementovali vlastním způsobem, než použili zde uvedené předvolené hodnoty.

**Poznámka:** IBM vám uděluje nevýhradní autorskou licenci na používání všech příkladů programových kódů, ze kterých můžete vygenerovat podobné funkce, které budou odpovídat vašim vlastním specifickým požadavkům. Všechny příklady kódů poskytuje IBM pouze pro účely ilustrace. Uvedené příklady nebyly důkladně testovány za všech podmínek. Z tohoto důvodu IBM nemůže zaručit nebo implikovat spolehlivost, funkčnost nebo provoz těchto programů. Všechny programy zde obsažené jsou vám nabízeny "tak jak jsou", bez záruky jakéhokoliv druhu. Výslovně se zříkáme všech odvozených záruk neporušení, prodejnosti nebo vhodnosti pro určitý účel.

Změňte tento úryvek kódu tak, aby odpovídal vašim požadavkům a potřebám na použití rozhraní Add Verifier API v předinstalačním programu výstupního bodu, které přidá požadovaný certifikát pro ověřování podpisů na zákazníkuv server iSeries během instalace vašeho produktu.

```
/* ----- */
/* */
/* 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 *   pathname 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;
}

```

Pokud jste provedli výše uvedené úlohy, můžete nyní balit vaši aplikaci a distribuovat ji k vašim zákazníkům. Během instalace vaší aplikace budou ověřeny podepsané aplikační objekty. Později mohou zákazníci používat produkt DCM (Digital Certificate Manager) k ověřování podpisů na vašich aplikačních objektech. Tímto způsobem umožníte vašim zákazníkům určit, zda je zdroj aplikace důvěryhodný, a také určit, zda byly po podepsání aplikace provedeny nějaké změny.

**Poznámka:** Váš instalační program mohl u vašeho zákazníka vytvořit paměť certifikátů \*SIGNATUREVERIFICATION s předvoleným heslem. Měli byste upozornit zákazníka, že by měl pomocí produktu DCM přenastavit heslo paměti certifikátů, co nejdříve to bude možné, aby tuto paměť certifikátů ochránil před neoprávněným přístupem.

#### **Krok 8: Přimějte zákazníky přenastavit předvolené heslo paměti certifikátů \*SIGNATUREVERIFICATION**

Rozhraní Add Verifier API může během procesu instalace produktu vytvořit paměť certifikátů \*SIGNATUREVERIFICATION na zákaznickově serveru iSeries. Pokud rozhraní API vytvořilo uvedenou paměť certifikátů, vytvořilo ji s předvoleným heslem. Proto byste měli upozornit zákazníky, aby pomocí produktu DCM přenastavili toto heslo, aby ochránili paměť certifikátů před neoprávněným přístupem.

Přimějte zákazníky, aby provedli níže uvedené kroky, vedoucí k vynulování hesla paměti certifikátů \*SIGNATUREVERIFICATION:

1. Spusťte produkt DCM.
2. V navigačním okně vyberte volbu **Výběr paměti certifikátů** a zvolte **\*SIGNATUREVERIFICATION** jako paměť certifikátů, kterou chcete otevřít.
3. Když se zobrazí stránka Paměť certifikátů a heslo, klepněte na **Vynulovat heslo** a zobrazí se stránka Nulování hesla paměti certifikátů.

**Poznámka:** Jestliže si nejste jisti, jak vyplnit určitý formulář v této řízené úloze, vyberte tlačítko s otazníkem (?) v horní části stránky, čímž se dostanete do online nápovědy.

4. Zadejte nové heslo paměti certifikátů, zadejte je ještě jednou, abyste je potvrdili, zvolte metodu ukončení platnosti hesla paměti certifikátů a klepněte na **Pokračovat**.

## **Scénář: Použití Centrální správy k podepisování objektů**

### **Situace**

Vaše společnost vyvíjí aplikace, které distribuuje na více serverů iSeries ve více lokalitách v rámci jedné společnosti. Jako správce sítě jste odpovědný za zajištění, že tyto aplikace jsou nainstalovány a aktualizovány na všech serverech iSeries vaší společnosti. V současné době používáte funkci Centrální správa produktu iSeries Navigator, abyste byli schopni jednoduše zabalit a distribuovat tyto aplikace a abyste byli schopni provádět další administrativní úkony, za které jste odpovědný. Přesto však strávíte více času, než byste chtěli, sledováním a řešením problémů těchto aplikací, protože dochází k neoprávněným změnám na jejich objektech. Z tohoto důvodu si přejete zabezpečit lépe integritu těchto objektů prostřednictvím jejich digitálních podpisů.

Pečlivě jste prozkoumal schopnosti podepisování objektů systému OS/400 a zjistil jste, že počínaje verzí V5R2 vám funkce Centrální správa umožňuje podepisovat objekty během jejich balení a distribuce. Prostřednictvím Centrální správy splníte cíle zabezpečení ochrany dat vaší společnosti efektivně a poměrně snadno. Dále jste se rozhodl, že vytvoříte lokálního vydavatele certifikátů (CA) a že ho budete používat k vydávání certifikátů pro podepisování objektů. Používání certifikátů vydávaných lokálním CA pro podepisování objektů snižuje náklady na používání této bezpečnostní technologie, protože nemusíte zakoupit certifikát od známého veřejného vydavatele certifikátů (CA).

Tento příklad slouží jako užitečný úvod do procesu konfigurace a používání podepisování objektů pro aplikace, které distribuujete na více serverů iSeries v rámci jedné společnosti.

### **Výhody scénáře**

Tento scénář má následující výhody:

- Použití Centrální správy k balení a podepisování objektů snižuje dobu, kterou musíte strávit nad distribucí podepsaných objektů na servery iSeries vaší společnosti.
- Použití Centrální správy k podepisování objektů v balíku programů snižuje počet nezbytných kroků, které musíte provést při podepisování objektů, protože proces podepisování objektů je součástí procesu balení.
- Podepisování objektů vám poskytuje prostředky ke snadnější kontrole, zda byly objekty po jejich podpisu změněny. Tak můžete snížit výskyt a řešení některých problémů, které by se objevily v budoucnosti při pátrání po problémech v aplikacích.
- Používání certifikátu, vydaného lokálním vydavatelem certifikátů (CA), k podepisování objektů snižuje náklady na implementaci.

## Cíle

V tomto scénáři si společnost MyCo, Inc. přeje digitálně podepisovat aplikace, které distribuuje na více serverů iSeries v rámci společnosti. Jako správce sítě společnosti MyCo, Inc. již používáte Centrální správu k řadě administrativních úloh na serveru iSeries. Proto jste se rozhodli rozšířit své současné použití Centrální správy o podepisování aplikací společnosti, které distribuujete na další servery iSeries.

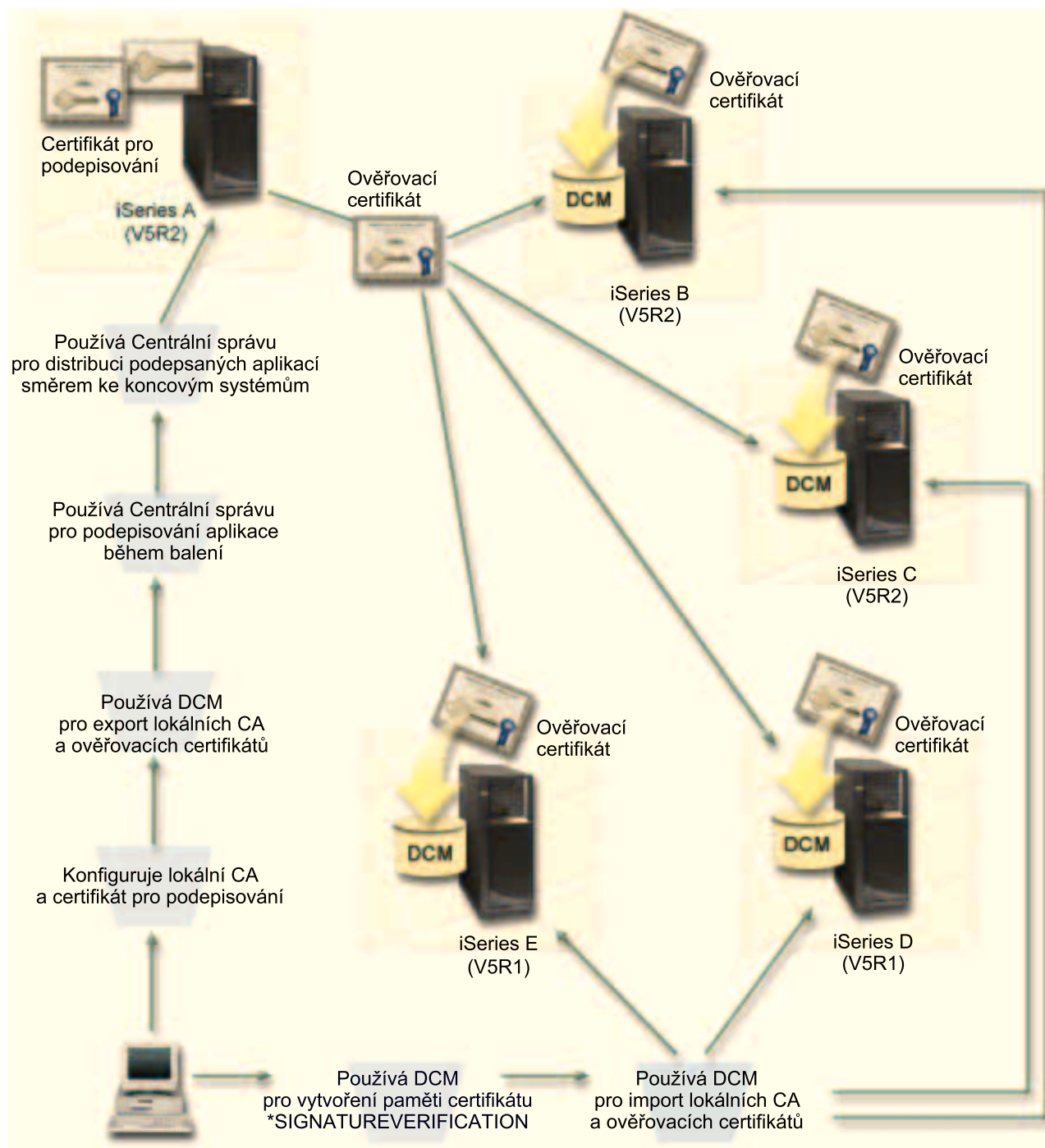
Cíle tohoto scénáře jsou následující:

- Firemní aplikace musí být podepsány certifikátem od lokálního vydavatele certifikátů (CA), aby byly sníženy náklady na aplikaci podepisování.
- Systémoví administrátoři a další určené uživatelé musí být schopni snadno ověřit digitální podpisy na všech serverech iSeries, aby ověřili zdroj a pravost firemních podepsaných objektů. Aby výše uvedené požadavky byly splněny, každý server iSeries musí mít vlastní kopii firemního certifikátu pro ověření podpisu a certifikátu lokálního vydavatele certifikátů (CA) ve své paměti certifikátů \*SIGNATUREVERIFICATION.
- Ověřením podpisů na firemních aplikacích mohou administrátoři iSeries a jiní určené uživatelé detekovat, zda byl obsah objektů změněn od doby jejich podpisu.
- Administrátoři musí být schopni používat Centrální správu k balení, podepisování, a poté také k distribuci jejich aplikací na své servery iSeries.

## Podrobnosti



Následující obrázek znázorňuje proces podepisování objektu a ověření podpisu, který bude implementován v tomto scénáři:



Obrázek zobrazuje následující body, které se vztahují k tomuto scénáři:

### Centrální systém (server iSeries A)

- Server iSeries A má nainstalován a spuštěn operační systém OS/400 verze 5, vydání 2 (V5R2).
- Server iSeries A slouží jako centrální systém, na kterém je spuštěna funkce Centrální správa, včetně balení a distribuce firemních aplikací.
- Server iSeries A má nainstalovány komponenty Cryptographic Access Provider 128-bit for iSeries (5722-AC3).

- Server iSeries A má nainstalovány a nakonfigurovány produkty Digital Certificate Manager (OS/400, volba 34) a IBM HTTP Server (5722–DG1).
- Server iSeries A vystupuje jako lokální vydavatel certifikátů (CA) a certifikát pro podepisování objektů je uložen v tomto systému.
- Server iSeries A je primárním systémem pro podepisování objektů pro aplikace společnosti. Podepisování produktových objektů, určených k distribuci zákazníkům, je prováděno na serveru A pomocí těchto úloh:
  1. Produkt DCM se použije k vytvoření lokálního vydavatele certifikátů (CA) a pomocí lokálního vydavatele certifikátů se vytvoří certifikát pro podepisování objektů.
  2. Produkt DCM se použije k vyexportování certifikátu CA a certifikátu pro ověřování podpisů do souboru, takže koncové systémy (servery iSeries B, C, D a E) mohou ověřovat podepsané objekty.
  3. Centrální správa se použije k podepisování aplikačních objektů a jejich balení spolu se soubory s certifikáty pro ověřování podpisů.
  4. Centrální správa se použije k distribuci podepsaných aplikací a certifikačních souborů na koncové systémy.

### Koncové systémy (servery iSeries B, C, D a E)

- Servery iSeries B a C mají nainstalován a spuštěn operační systém OS/400 verze 5, vydání 2 (V5R2).
- Servery iSeries D a E mají nainstalován a spuštěn operační systém OS/400 verze 5, vydání 1 (V5R1).
- Servery iSeries B, C, D a E mají nainstalovány a nakonfigurovány produkty Digital Certificate Manager (volba 34) a IBM HTTP Server (5722–DG1).
- Servery iSeries B, C, D a E obdržely z centrálního systému (server iSeries A) jak kopii certifikátu pro ověřování podpisů, tak i certifikátu lokálního CA, když tyto systémy obdržely podepsané aplikace.
- Produkt DCM byl použit k vytvoření paměti certifikátů \*SIGNATUREVERIFICATION a k naimportování certifikátu lokálního CA a certifikátu pro ověřování podpisů do této paměti certifikátů.

### Nezbytné podmínky a předpoklady

Nezbytné podmínky a předpoklady pro realizaci uvedeného scénáře jsou tyto:

1. Všechny servery iSeries splňují požadavky pro nainstalování a použití produktu DCM (Digital Certificate Manager).
2. Na žádném ze serverů iSeries dosud nikdo nekonfiguroval a nepoužíval produkt DCM.
3. Server iSeries A splňuje požadavky pro nainstalování a použití produktů iSeries Navigator a Centrální správa.
4. Server Centrální správy musí být spuštěn na všech koncových systémech iSeries.
5. Všechny servery iSeries mají nainstalovány nejnovější úroveň licencovaného programu Cryptographic Access Provider 128-bit (5722-AC3).
6. Systémová hodnota QVFYOBJRST Verify object signatures during restore) na všech serverech iSeries ve scénáři je předvoleně nastavena na hodnotu 3 a toto nastavení nebude změněno. Předvolené nastavení zajišťuje, aby server mohl ověřovat podpisy objektů, zatímco vy obnovujete podepsané objekty.
7. Správce sítě serveru iSeries A musí mít zvláštní oprávnění \*ALLOBJ, aby mohl podepisovat objekty, nebo jeho uživatelský profil musí mít oprávnění k aplikaci pro podepisování objektů.
8. Správce sítě nebo jiný uživatel, který vytváří paměť certifikátů v DCM, musí mít zvláštní oprávnění \*SECADM a \*ALLOBJ.
9. Systémoví administrátoři a/nebo jiní uživatelé dalších serverů iSeries musí mít zvláštní oprávnění \*AUDIT, aby byli schopni ověřovat podpisy objektů.

### Kroky scénáře

Při realizaci tohoto scénáře musíte provést dvě skupiny úloh: V první sadě úloh budete konfigurovat server iSeries A tak, aby používal Centrální správu k podepisování a distribuci aplikací. Druhá sada úloh umožní administrátorům systému a jiným určeným uživatelům ověřovat podpisy na těchto aplikacích na všech ostatních serverech iSeries.

### Skupina úloh týkající se podepisování objektů

Musíte dokončit každou z níže uvedených úloh na serveru iSeries A, abyste byli schopni podepisovat objekty způsobem, který je popsán ve scénáři:

1. Dokončíte všechny kroky týkající se nezbytných předpokladů, které se vztahují k instalaci a konfiguraci potřebných produktů iSeries.
2. Pomocí produktu DCM (Digital Certificate Manager) vytvoříte lokálního vydavatele certifikátů (CA), který vydá soukromý certifikát pro podepisování objektů.
3. Pomocí produktu DCM vytvoříte definici aplikace.
4. Pomocí produktu DCM přiřadíte certifikát k definici aplikace pro podepisování objektů.
5. Pomocí produktu DCM vyexportujete certifikáty, které musí používat další systémy k ověřování podpisů objektů. Musíte vyexportovat jak kopii certifikátu lokálního CA, tak i kopii certifikátu pro podepisování objektů do souboru. Obě kopie tvoří certifikát pro ověření podpisu.
6. Přeneste certifikační soubory na všechny koncové systémy iSeries, na kterých zamýšlíte provádět ověřování podpisů.
7. Pomocí Centrální správy podepište aplikační objekty.

### Skupina úloh týkající se ověřování podpisů

Měli byste dokončit každou z níže uvedených úloh v každém koncovém systému iSeries, než začnete používat Centrální správu k přenesení podepsaných aplikačních objektů na tyto koncové systémy. Konfiguraci ověřování podpisů musíte dokončit, abyste mohli úspěšně ověřovat podpisy během obnovování podepsaných objektů na koncových systémech.

V každém koncovém systému iSeries musíte provést níže uvedené úlohy, abyste byli schopni ověřovat podpisy na objektech způsobem, který je popsán ve scénáři:

8. Pomocí produktu DCM (Digital Certificate Manager) vytvoříte paměť certifikátů \*SIGNATUREVERIFICATION.
9. Pomocí produktu DCM naimportujte certifikát lokálního vydavatele certifikátů (CA) a certifikát pro ověření podpisu.

### Podrobnosti konfigurace

Chcete-li nakonfigurovat a používat Centrální správu k podepisování objektů způsobem, který je popsán v tomto scénáři, postupujte takto:

#### Krok 1: Dokončíte všechny kroky týkající se nezbytných předpokladů

K tomu, abyste mohli začít s konfiguračními úlohami při realizaci tohoto scénáře, musíte nejprve splnit všechny nezbytné předpoklady týkající se instalace a konfigurace potřebných produktů na serverech iSeries.

#### >Krok 2: Vytvoříte lokálního vydavatele certifikátů (CA), který bude vydávat soukromý certifikát pro podepisování objektů

Pokud chcete použít produkt DCM (Digital Certificate Manager) k vytvoření lokálního vydavatele certifikátů (CA), budete muset vyplnit řadu formulářů. Tyto formuláře vás provedou procesem vytvoření lokálního CA a dalšími úlohami potřebnými k zahájení používání digitálních certifikátů pro SSL (Secure Sockets Layer),

podepisování objektů a ověřování podpisů. I když v tomto scénáři nemusíte nakonfigurovat certifikáty pro SSL, musíte vyplnit všechny formuláře uvedené v úloze, abyste mohli nakonfigurovat systém pro podepisování objektů.

Chcete-li pomocí produktu DCM vytvořit a provozovat lokálního CA, postupujte následovně:

1. Spusťte produkt DCM.
2. V navigačním okně produktu DCM vyberte volbu **Vytvoření vydavatele certifikátů (CA)**. Zobrazí se sada formulářů.

**Poznámka:** Jestliže si nejste jisti, jak vyplnit určitý formulář v této řízené úloze, vyberte tlačítko s otázkou (?) v horní části stránky, čímž se dostanete do online nápovědy.

3. Vyplňte všechny formuláře pro tuto řízenou úlohu. Během provádění této úlohy musíte:
  - a. Poskytnout identifikační informace pro lokálního CA.
  - b. Nainstalovat certifikát lokálního CA do vašeho prohlížeče, aby váš software mohl rozpoznat lokálního CA a potvrzovat certifikáty, které lokální CA vydá.
  - c. Zadat strategická data pro vašeho lokálního CA.
  - d. Pomocí nového lokálního CA vydat serverový nebo klientský certifikát, který budou vaše aplikace používat pro připojení SSL.

**Poznámka:** Přestože tento scénář nepoužívá tento typ certifikátu, musíte jej vytvořit, abyste mohli pomocí lokálního CA vydávat certifikát pro podepisování objektů, který potřebujete. Pokud zrušíte úlohu, aniž byste vytvořili tento certifikát, musíte samostatně vytvořit váš certifikát pro podepisování objektů a paměť certifikátů \*OBJECTSIGNING, ve které bude certifikát pro podepisování objektů uložen.

- e. Vyberete aplikace, které mohou použít serverový nebo klientský certifikát pro připojení SSL.

**Poznámka:** Pro účely tohoto scénáře nemusíte zvolit žádnou aplikaci a klepnutím na tlačítko **Pokračovat** zobrazíte další formulář.

- f. Pomocí nového lokálního CA vydat certifikát pro podepisování objektů, který budou používat aplikace k digitálnímu podepisování objektů. Tato podúloha vytvoří paměť certifikátů \*OBJECTSIGNING. Tuto paměť certifikátů budete používat při správě certifikátů pro podepisování objektů.
- g. Vyberete aplikace, které by měly důvěřovat vašemu lokálnímu CA.

**Poznámka:** Pro účely tohoto scénáře nemusíte zvolit žádnou aplikaci a klepnutím na tlačítko **Pokračovat** dokončíte úlohu.

Teď, když jste vytvořili lokálního vydavatele certifikátů (CA) a certifikát pro podepisování objektů, musíte definovat aplikaci pro podepisování objektů, která bude uvedený certifikát pro podepisování objektů používat.

### Krok 3: Vytvořte definici aplikace pro podepisování objektů

Když jste vytvořili váš certifikát pro podepisování objektů, musíte pomocí produktu DCM (Digital Certificate Manager) definovat aplikaci pro podepisování objektů, která uvedený certifikát bude používat k podepisování objektů. Definice aplikace se nemusí odkazovat na skutečnou aplikaci. Definice aplikace, kterou vytvoříte, by měla místo toho popisovat typ nebo skupinu objektů, které hodláte podepisovat. Definici potřebujete, abyste mohli přiřadit ID aplikace k certifikátu a tak aktivovali proces podepisování.

Chcete-li pomocí produktu DCM vytvořit definici aplikace pro podepisování objektů, postupujte takto:

1. V navigačním okně klepněte na volbu **Výběr paměti certifikátů** a zvolte \*OBJECTSIGNING jako paměť certifikátů, kterou chcete otevřít.
2. Když se zobrazí stránka Paměť certifikátů a heslo, zadejte heslo, které jste pro danou paměť certifikátů uvedli, když jste ji tvořili, a klepněte na **Pokračovat**.

3. V navigačním okně vyberte volbu **Správa aplikací**. Zobrazí se seznam úloh.
4. Vyberte ze seznamu úloh volbu **Přidat aplikaci**. Zobrazí se formulář pro definici aplikace.
5. Vyplňte formulář a klepněte na **Přidat**.

Nyní musíte přiřadit váš certifikát pro podepisování objektů k aplikaci, kterou jste vytvořili.

#### **Krok 4: Přiřaďte certifikát k definici aplikace pro podepisování objektů**

Chcete-li přiřadit certifikát k aplikaci pro podepisování objektů, proveďte následující kroky:

1. V navigačním okně DCM vyberte volbu **Správa certifikátů**. Zobrazí se seznam úloh.
2. Ze seznamu úloh vyberte volbu **Přiřadit certifikát**. Zobrazí se seznam certifikátů v aktuální paměti certifikátů.
3. Ze seznamu vyberte příslušný certifikát a klepněte na volbu **Přiřadit k aplikacím**. Zobrazí se seznam definic aplikací pro aktuální paměť certifikátů.
4. Vyberte ze seznamu jednu nebo více aplikací a klepněte na **Pokračovat**. Zobrazí se stránka, která buď potvrdí přiřazení certifikátu, nebo v případě problémů informuje o chybách.

Až dokončíte tuto úlohu, budete připraveni pomocí Centrální správy podepisovat objekty během jejich balení a distribuce. Pokud však chcete zajistit, abyste vy i jiní určené uživatele mohli ověřovat podpisy, musíte vyexportovat nezbytné certifikáty do souboru a přenést je na všechny koncové systémy iSeries. Musíte také dokončit všechny úlohy konfigurace ověřování podpisů v každém koncovém systému iSeries, než budete moci používat Centrální správu k přenášení podepsaných aplikačních objektů na tyto koncové systémy. Konfiguraci ověřování podpisů musíte dokončit, abyste mohli úspěšně ověřovat podpisy během obnovování podepsaných objektů na koncových systémech.

#### **Krok 5: Vyexportujte certifikáty, abyste povolili ověřování podpisů v ostatních systémech iSeries**

Použití metody podepisování objektů k zajištění integrity obsahu vyžaduje, abyste vy a další určené uživatele měli k dispozici prostředky k ověřování pravosti podpisu. K ověřování podpisů objektů ve stejném systému, který tyto objekty podepisuje, musíte pomocí produktu DCM vytvořit paměť certifikátů \*SIGNATUREVERIFICATION. Uvedená paměť certifikátů musí obsahovat jak kopii certifikátu pro podepisování objektů, tak i kopii certifikátu CA pro CA, který vydal certifikát pro podepisování.

Chcete-li umožnit i jiným, aby mohli ověřovat podpisy, musíte také jim poskytnout kopii certifikátu, který podepisuje objekty. Pokud používáte lokálního vydavatele certifikátů (CA) k vydávání certifikátů, musíte těmto určeným uživatelům poskytnout také kopii certifikátu lokálního CA.

Jestliže si přejete pomocí produktu DCM ověřovat podpisy ve stejném systému, který podepsal objekty (server iSeries A v tomto scénáři), proveďte následující kroky:

1. V navigačním okně vyberte volbu **Vytvoření nové paměti certifikátů** a zvolte \*SIGNATUREVERIFICATION jako paměť certifikátů, která se má vytvořit.
2. Klepnutím na **Ano** zkopírujete existující certifikáty pro podepisování objektů do nové paměti certifikátů jako certifikáty pro ověřování podpisů.
3. Uveďte heslo pro novou paměť certifikátů a klepněte na **Pokračovat**, abyste vytvořili paměť certifikátů. Nyní můžete prostřednictvím DCM ověřovat podpisy objektů ve stejném systému, který používáte pro podepisování objektů.

Pokud chcete pomocí produktu DCM vyexportovat kopii certifikátu lokálního CA a kopii certifikátu pro podepisování objektů jako certifikát pro ověřování podpisů, abyste mohli ověřovat podpisy objektů v jiných systémech, proveďte následující kroky:

1. V navigačním okně vyberte volbu **Správa certifikátů** a pak úlohu **Export certifikátu**.

2. Vyberte volbu **Vydavatel certifikátu (CA)** a klepněte na **Pokračovat**. Zobrazí se seznam certifikátů CA, které můžete vyexportovat.
3. Vyberte ze seznamu certifikát lokálního CA, kterého jste vytvořili dříve, a klepněte na **Export**.
4. Zadejte **Soubor** jako vaše místo určení exportu a klepněte na **Pokračovat**.
5. Zadejte úplnou cestu a jméno souboru exportovaného certifikátu lokálního CA a klepněte na **Pokračovat**, abyste vyexportovali certifikát.
6. Klepnutím na **OK** ukončíte stránku pro potvrzení exportu. Nyní můžete vyexportovat kopii certifikátu pro podepisování objektů.
7. Znovu vyberte úlohu **Export certifikátu**.
8. Vyberte **Podepisování objektů** a zobrazí se seznam certifikátů pro podepisování objektů, které můžete vyexportovat.
9. Vyberte ze seznamu odpovídající certifikát pro podepisování objektů a klepněte na **Export**.
10. Zvolte **Soubor jako certifikát pro ověřování podpisů** jako místo určení a klepněte na **Pokračovat**.
11. Zadejte úplnou cestu a jméno souboru exportovaného certifikátu pro ověřování podpisů a klepněte na **Pokračovat**, abyste vyexportovali certifikát.

Nyní můžete tyto soubory přenést na koncové systémy iSeries, na kterých chcete ověřovat podpisy, jež jste vytvořili pomocí certifikátu.

#### Krok 6: Přeneste soubory certifikátů na koncové systémy iSeries

Soubory certifikátů, které jste vytvořili na serveru iSeries A, musíte přenést na koncové systémy iSeries, abyste je mohli nakonfigurovat pro ověřování objektů, které jste podepsali. K přenesení souborů certifikátů můžete použít několik různých metod. K přenesení souborů můžete například použít FTP (File Transfer Protocol) nebo distribuci sad programů v Centrální správě.

#### Krok 7: Podepište objekty pomocí Centrální správy

Proces podepisování objektů pomocí Centrální správy je součástí distribučního procesu balení softwaru. Musíte dokončit všechny úlohy konfigurace ověřování podpisů v každém koncovém systému iSeries, než budete moci používat Centrální správu k přenášení podepsaných aplikačních objektů na tyto koncové systémy. Konfiguraci ověřování podpisů musíte dokončit, abyste mohli úspěšně ověřovat podpisy během obnovování podepsaných objektů na koncových systémech.

Jestliže chcete podepisovat aplikace, které distribuujete na koncové systémy iSeries, způsobem, který je popsán v tomto scénáři, postupujte takto:

1. Používejte Centrální správu k balení a distribuci softwarových produktů.
2. Až se v **průvodci definicí produktů** objeví panel **Identifikace**, klepněte na tlačítko **Rozšířené**. Zobrazí se panel **Rozšířená identifikace**.
3. Do pole **Digitální podpisy** zadejte ID aplikace pro aplikaci pro podepisování objektů, kterou jste vytvořili dříve a klepněte na **OK**.
4. Dokončete průvodce a pokračujte v procesu balení a distribuce softwarových produktů pomocí Centrální správy.

#### Krok 8: Úlohy ověřování podpisů: Vytvořte paměť certifikátů \*SIGNATUREVERIFICATION na koncových systémech iSeries

Jestliže chcete ověřovat podpisy v koncových systémech iSeries (jak je uvedeno v tomto scénáři), každý systém musí mít kopii odpovídajících certifikátů pro ověřování podpisů ve své paměti certifikátů \*SIGNATUREVERIFICATION. Pokud objekty podepsal soukromý certifikát, tato paměť certifikátů musí také obsahovat kopii certifikátu lokálního vydavatele certifikátů (CA).

Chcete-li vytvořit paměť certifikátů \*SIGNATUREVERIFICATION, postupujte následovně:

1. Spusťte produkt DCM.
2. V navigačním okně produktu DCM vyberte volbu **Vytvoření nové paměti certifikátů** a zvolte **\*SIGNATUREVERIFICATION** jako paměť certifikátů, která se má vytvořit.

**Poznámka:** Jestliže si nejste jisti, jak vyplnit určitý formulář v této řízené úloze, vyberte tlačítko s otázníkem (?) v horní části stránky, čímž se dostanete do online nápovědy.

3. Uveďte heslo pro novou paměť certifikátů a klepněte na **Pokračovat**, abyste vytvořili paměť certifikátů. Nyní můžete naimportovat certifikáty do paměti certifikátů a používat je k ověřování podpisů objektů.

#### **Krok 9: Úlohy ověřování podpisů: Naimportujte certifikáty**

Pokud chcete ověřit podpis na objektu, paměť certifikátů \*SIGNATUREVERIFICATION musí obsahovat kopii certifikátu pro ověřování podpisů. Jestliže je certifikát pro podepisování soukromým certifikátem, paměť certifikátů musí také obsahovat kopii certifikátu lokálního vydavatele certifikátů (CA), který vydal certifikát pro podepisování. V tomto scénáři byly vyexportovány oba certifikáty do souboru a tento soubor byl přenesen na každý koncový systém iSeries.

Chcete-li naimportovat tyto certifikáty do paměti certifikátů \*SIGNATUREVERIFICATION, postupujte následovně:

1. V navigačním okně produktu DCM klepněte na **Výběr paměti certifikátů** a jako paměť certifikátů, kterou chcete otevřít, zvolte **\*SIGNATUREVERIFICATION**.
2. Když se zobrazí stránka Paměť certifikátů a heslo, zadejte heslo, které jste pro danou paměť certifikátů uvedli, když jste ji tvořili, a klepněte na **Pokračovat**.
3. Když se navigační okno obnoví, vyberte volbu **Správa certifikátů**. Zobrazí se seznam úloh.
4. Ze seznamu úloh vyberte úlohu **Import certifikátu**.
5. Jako typ certifikátu, který budete importovat, vyberte **Vydavatel certifikátu (CA)** a klepněte na **Pokračovat**.

**Poznámka:** Musíte nejprve naimportovat certifikát lokálního CA a teprve poté soukromý certifikát pro ověřování podpisů, jinak import certifikátu pro ověřování podpisů selže.

6. Uveďte úplnou cestu a jméno souboru s certifikátem CA a klepněte na **Pokračovat**. Zobrazí se zpráva, která buď potvrdí úspěšnost importu certifikátu, nebo poskytne informace o chybách v případě, že import selhal.
7. Znovu vyberte úlohu **Import certifikátu**.
8. Jako typ certifikátu, který se má naimportovat, zvolte **Ověřování podpisů** a klepněte na **Pokračovat**.
9. Uveďte úplnou cestu a jméno souboru souboru s certifikátem pro ověřování podpisů a klepněte na **Pokračovat**. Zobrazí se zpráva, která buď potvrdí úspěšnost importu certifikátu, nebo poskytne informace o chybách v případě, že import selhal.

Váš systém iSeries je nyní schopen ověřovat podpisy na objektech, které jste vytvořili s odpovídajícím certifikátem pro podepisování, během obnovování těchto podepsaných objektů.

---

## **Koncepce podepisování objektů**

Než začnete používat funkce systému iSeries v oblasti podepisování objektů a ověřování podpisů, může být pro vás prospěšné projít si některé z těchto koncepcí:

### **Digitální podpisy**

Zde se dozvíte, co jsou to digitální podpisy a jakou úroveň ochrany poskytují.

### **Podpisovatelné objekty**

Zde se dozvíte, jaké objekty iSeries můžete podepsat a jaké jsou volby podpisu objektů typu command (\*CMD).

### **Zpracování podepisování objektů**

Zde se dozvíte, jak probíhá proces podepisování objektů a jaké parametry můžete pro tento proces nastavit.

### **Zpracování ověřování podpisů**

Zde se dozvíte, jak probíhá proces ověřování podpisů a jaké parametry můžete pro tento proces nastavit.

## **Digitální podpisy**

Operační systém OS/400 poskytuje podporu pro používání certifikátů k digitálnímu "podepisování" objektů. Digitální podpis na objektu je vytvořen za použití určité formy kryptografie a je ekvivalentem osobního podpisu na psaném dokumentu. Digitální podpis poskytuje důkaz o původu objektu a prostředek ověření integrity objektu. Vlastník digitálního certifikátu "podepisuje" objekt pomocí soukromého klíče certifikátu. Příjemce objektu použije odpovídající veřejný klíč certifikátu při dešifrování podpisu, který ověřuje integritu podepsaného objektu a ověřuje odesílatele jako zdroj objektu.

Podpora pro podepisování objektů posiluje tradiční systémové nástroje systému iSeries pro řízení toho, kdo může měnit objekty. Tradiční způsoby řízení nemohou objekt chránit před neoprávněným narušením v době, kdy se objekt přenáší přes Internet nebo jinou nedůvěryhodnou síť. Protože můžete zjistit, zda byl obsah objektu změněn od okamžiku jeho podpisu, můžete snadněji určit, zda máte věřit objektům, získaným podobným způsobem.

Digitální podpis je zašifrovaný matematický součet dat v objektu. Digitálním podpisem nedojde k zašifrování objektu a jeho obsahu a k zajištění jejich privátnosti, avšak samotný součet je zašifrován, a zabraňuje tak neoprávněným změnám v jeho hodnotě. Každý, kdo se chce ujistit, že objekt nebyl v průběhu přenosu změněn a že pochází ze schváleného, legálního zdroje, může použít veřejný klíč podpisového certifikátu pro ověření originálního digitálního podpisu. Pokud již podpis neodpovídá, mohla být data změněna. V takovém případě může příjemce odmítnout objekt přijmout a požádat podepisovatele objektu o zaslání další kopie objektu.

Podpis na určitém objektu reprezentuje systém, který objekt podepsal, nikoliv konkrétního uživatele v rámci tohoto systému (i když uživatel musí mít příslušné oprávnění, aby mohl používat certifikáty pro podepisování objektů).

Jestliže dojdete k závěru, že používání digitálních podpisů vyhovuje vašim potřebám a strategiím v oblasti zabezpečení ochrany dat, měli byste dále vyhodnotit, zda používat veřejné certifikáty nebo vydávat soukromé certifikáty. Hodláte-li distribuovat objekty uživatelům z řad široké veřejnosti, měli byste uvažovat o použití certifikátů pro podepisování objektů od některého známého veřejného vydavatele certifikátu (CA). Použití veřejných certifikátů zajišťuje, že ostatní mohou snadno a levně ověřovat podpisy, které na objekty, jež jim distribuujete, umístíte. Jestliže však hodláte distribuovat objekty výhradně uvnitř vaší organizace, může být výhodnější používat produkt DCM (Digital Certificate Manager) k provozování vlastního lokálního vydavatele certifikátů (CA), který bude vydávat certifikáty pro podepisování objektů. Použití soukromých certifikátů od lokálního CA je levnější varianta, než nakupování certifikátů od známého veřejného CA.

### **Typy digitálních podpisů**

Počínaje verzí V5R2 můžete podepisovat také objekty typu command (\*CMD). Můžete si také vybrat jeden ze dvou typů podpisů pro objekty \*CMD: podpisy jádra objektu nebo podpisy celého objektu.

- **Podpisy celých objektů**

Tento typ podpisu pokrývá celý objekt, kromě několika nedůležitých bajtů objektu.



- **Podpisy jádra objektů**

Tento typ podpisu pokrývá životně důležité bajty objektu \*CMD. Tento podpis však nepokrývá takové bajty, které jsou předmětem nejčastějších změn. Uvedený typ podpisu umožňuje provádění některých změn příkazu, aniž by bylo nutné ověřovat podpis. Které bajty objektu nejsou pokryty při použití typu podpisu jádra objektu se liší podle objektu \*CMD. Podpisy jádra nepokrývají například předvolby parametrů na objektech \*CMD. Příklady změn, které nevyžadují ověření podpisu při použití podpisu jádra objektu zahrnují:

- Změny předvoleb příkazu.
- Přidání programu pro kontrolu platnosti k příkazu, který takový program dosud nemá.
- Změna parametru Where allowed to run (Kde je povoleno spustit).
- Změna parametru Allow limited users (Povolit omezené uživatele).

Chcete-li se dozvědět více o tom, které objekty iSeries můžete podepisovat a které bajty objektů \*CMD pokrývá podpis jádra objektu, prostudujte si část Podepisovatelné objekty.

## **Podepisovatelné objekty**

Digitálně podepsat můžete řadu typů objektů operačního systému OS/400, bez ohledu na způsob, který použijete k jejich podpisu. Můžete podepsat libovolný objekt (\*STMF), který uložíte v integrovaném systému souborů daného systému, s výjimkou objektů uložených v knihovnách. Pokud má objekt připojen Java program, bude tento program také podepsán. Můžete podepsat pouze tyto objekty v systému souborů QSYS.LIB: programy (\*PGM), servisní programy (\*SRVPGM), moduly (\*MODULE), SQL balíky (\*SQLPKG), \*FILE (pouze soubory typu save) a příkazy (\*CMD).

Abyste mohli podepsat objekt, musí být v paměti lokálního systému. Když například provozujete Windows 2000 server na serveru Integrated xSeries Server for iSeries, máte v integrovaném systému souborů k dispozici systém souborů QNTC. Adresáře v tomto systému souborů se nepovažují za lokální, protože obsahují soubory, které jsou vlastněny operačním systémem Windows 2000. Nemůžete také podepsat prázdné objekty nebo objekty, které jsou kompilovány pro dřívější verzi než V5R1.

### **Podpisy objektů \*CMD**

Při podepisování objektů \*CMD si můžete zvolit jeden ze dvou typů podpisů, které je možné použít na objektu \*CMD. Můžete se rozhodnout, že budete podepisovat celý objekt, nebo že budete podepisovat pouze část jádra objektu. Když se rozhodnete podepsat celý objekt, podpis se použije na celý objekt s výjimkou několika nepodstatných bajtů. Podpis celého objektu pokrývá položky, obsažené v podpisu jádra objektu.

Pokud se rozhodnete podepsat pouze jádro objektu, budou podstatné bajty chráněny podpisem, zatímco bajty, které se častěji mění, podepsány nebudou. To, které bajty nebudou podepsány, se liší v závislosti na objektu \*CMD, ale mohou to být např. bajty, které určují režim, v jakém je objekt platný, bajty, které určují, kde je objektu povoleno pracovat. Podpisy jádra nepokrývají například předvolby parametrů na objektech \*CMD. Tento typ podpisu umožňuje provádět některé změny příkazu, aniž by bylo nutné ověřovat jeho podpis. Příklady změn, které nevyžadují ověření podpisu při použití podpisu jádra objektu zahrnují:

- Změny předvoleb příkazu.
- Přidání programu pro kontrolu platnosti k příkazu, který takový program dosud nemá.
- Změna parametru Where allowed to run (Kde je povoleno spustit).
- Změna parametru Allow limited users (Povolit omezené uživatele).

Níže uvedená tabulka popisuje přesně, které bajty v objektu \*CMD jsou součástí podpisu jádra objektu.

### **Složení podpisu jádra objektu u objektů \*CMD**

<b>Část objektu</b>	<b>Vztah k podpisu jádra objektu</b>
Předvolby příkazu, měněné příkazem CHGCMDDFT	Není součástí podpisu jádra objektu
Program, který zpracovává příkaz a knihovnu	Je vždy součástí podpisu jádra objektu
Zdrojový soubor a knihovna REXX	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Člen zdroje REXX	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Prostředí a knihovna příkazu REXX	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Jméno programu výstupního bodu, knihovna a kód výstupního bodu REXX	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Program a knihovna kontroly platnosti	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Režim, ve kterém je platný	Není součástí podpisu jádra objektu
Kde je povoleno spustit	Není součástí podpisu jádra objektu
Povolit omezené uživatele	Není součástí podpisu jádra objektu
Přihrádka nápovědy	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Skupina dialogových oken a knihovna nápovědy	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Identifikátor nápovědy	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Index vyhledávání a knihovna nápovědy	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Aktuální knihovna	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Knihovna produktů	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Knihovna a program pro přepsání názvem	Je součástí podpisu jádra objektu, pokud je u příkazu uveden v okamžiku podepisování, jinak součástí podpisu jádra objektu není
Text (popis)	Není součástí ani podpisu jádra objektu, ani podpisu celého objektu, protože není uložen v objektu
Povolit GUI (graphical user interface)	Není součástí podpisu jádra objektu

## Zpracování podepisování objektů

Když podepisujete objekty, můžete uvést následující volby, vztahující se ke zpracování podepisování objektů.

- **Zpracování chyby**

Můžete uvést, jaký typ zpracování chyby by aplikace měla používat při vytváření podpisů na více než jednom objektu. Můžete zadat, že aplikace má zastavit podepisování objektů, vyskytne-li se chyba, nebo že má pokračovat v podepisování dalších objektů v procesu.

- **Duplikace podpisu objektu**

Můžete uvést, jak by měla aplikace postupovat v procesu podepisování, jestliže se aplikace pokouší znovu podepsat objekt. Můžete zadat, zda se má na objektu ponechat originální podpis, nebo zda se má originální podpis nahradit novým podpisem.

- **Objekty v podadresářích**

Můžete uvést, jak by aplikace měla postupovat při podepisování objektů v podadresářích. Můžete zadat, že aplikace individuálně podepíše objekty ve všech podadresářích, nebo že aplikace podepíše pouze objekty v hlavním adresáři a všechny podadresáře bude ignorovat.

- **Rozsah podpisu objektu**

Při podepisování objektů \*CMD můžete uvést, zda se bude podepisovat celý objekt nebo zda se bude podepisovat pouze jádro objektu.

## Zpracování ověřování podpisů

Při ověřování podpisů můžete uvést následující volby, vztahující se ke zpracování ověřování podpisů.

- **Zpracování chyby**

Můžete uvést, jaký typ zpracování chyby by aplikace měla používat při ověřování podpisů na více než jednom objektu. Můžete zadat, že aplikace má zastavit ověřování podpisů, vyskytne-li se chyba, nebo že má pokračovat v ověřování podpisů dalších objektů v procesu.

- **Objekty v podadresářích**

Můžete uvést, jak by aplikace měla postupovat při ověřování podpisů objektů v podadresářích. Můžete zadat, že aplikace individuálně ověří podpisy na objektech ve všech podadresářích, nebo že aplikace ověří podpisy pouze na objektech v hlavním adresáři a všechny podadresáře bude ignorovat.

- **Ověřování podpisu jádra objektu nebo celého objektu**

Existují systémová pravidla, která určují, jak má systém pracovat s podpisy jádra nebo celého objektu během procesu ověřování podpisů. Tato pravidla jsou následující:

- Pokud na objektu neexistuje žádný podpis, proces ověřování podpisu ohlásí, že objekt není podepsán, a bude pokračovat v ověřování dalších objektů v procesu.
- Pokud byl objekt podepsán systémem ověřeným zdrojem (IBM), podpis musí odpovídat, jinak proces ověřování selže. Pokud podpis odpovídá, proces ověřování pokračuje. Podpis je zašifrovaný matematický součet dat v objektu. Z tohoto důvodu se považuje, že podpis odpovídá, pokud data v objektu během ověřování odpovídají datům v objektu ve chvíli, kdy byl podepsán.
- Pokud má objekt nějaký podpis celého objektu, který je ověřený (na základě certifikátu v paměti certifikátů \*SIGNATUREVERIFICATION), minimálně jeden z těchto podpisů musí odpovídat, jinak proces ověřování podpisů selže. Pokud alespoň jeden podpis celého objektu odpovídá, proces ověřování podpisů pokračuje.
- Pokud má objekt nějaký podpis jádra objektu, který je ověřený, minimálně jeden z nich musí odpovídat certifikátu v paměti certifikátů \*SIGNATUREVERIFICATION, jinak proces ověřování podpisů selže. Pokud alespoň jeden podpis jádra objektu odpovídá, proces ověřování podpisů pokračuje.

---

## Nezbytné předpoklady pro podepisování objektů a ověřování podpisů

Schopnosti operačního systému OS/400 v oblasti podepisování objektů a ověřování podpisů vám poskytují další výkonné prostředky řízení objektů na vašem serveru iSeries. Chcete-li plně využívat výhod těchto schopností, musíte splnit nezbytné předpoklady pro jejich používání.

### Nezbytné předpoklady pro podepisování objektů

Existuje mnoho způsobů, které můžete používat k podepisování objektů. Jejich použití závisí na vašich obchodních a bezpečnostních potřebách.

- Můžete používat produkt DCM (Digital Certificate Manager).
- Můžete napsat program, který bude používat rozhraní Sign Object API.
- Můžete používat funkci Centrální správa produktu iSeries Navigator, abyste podepisovali objekty během jejich balení za účelem distribuce na koncové systémy iSeries.

To, pro který z výše uvedených způsobů se rozhodnete, závisí na vašich obchodních potřebách a požadavcích v oblasti zabezpečení ochrany dat. Bez ohledu na způsob, který hodláte používat k podepisování objektů, musíte zajistit, že budou splněny některé nezbytné předpoklady:

- Musíte splnit nezbytné předpoklady pro instalaci a použití produktu DCM (Digital Certificate Manager).
  - K vytvoření paměti certifikátů \*OBJECTSIGNING musíte použít produkt DCM. Tuto paměť certifikátů vytvoříte buď v procesu vytváření lokálního vydavatele certifikátů (CA), nebo v procesu správy certifikátů pro podepisování objektů od veřejného vydavatele certifikátů (CA).
  - Paměť certifikátů \*OBJECTSIGNING musí obsahovat minimálně jeden certifikát, a to buď ten, který jste vytvořili s pomocí lokálního vydavatele certifikátů (CA), nebo ten, který jste získali od veřejného vydavatele certifikátů (CA).
  - Musíte použít produkt DCM k vytvoření minimálně jedné definice aplikace pro podepisování objektů, kterou budete používat k podepisování objektů.
  - Musíte použít produkt DCM k přiřazení určitého certifikátu k definici aplikace pro podepisování objektů.
- Uživatelský profil iSeries, který podepisuje objekty, musí mít speciální oprávnění \*ALLOBJ. Uživatelský profil iSeries, který vytváří paměť certifikátů \*SIGNATUREVERIFICATION, musí mít speciální oprávnění \*SECADM a \*ALLOBJ.

#### Nezbytné předpoklady pro ověřování podpisů

Existuje mnoho způsobů, které můžete používat při ověřování podpisů na objektech:

- Můžete používat produkt DCM (Digital Certificate Manager).
- Můžete napsat program, který bude používat rozhraní Verify Object (QYDOVFYO) API.
- Můžete použít různé příkazy, jako např. příkaz CHKOBJITG (Check Object Integrity).

To, pro který z výše uvedených způsobů se rozhodnete, závisí na vašich obchodních potřebách a požadavcích v oblasti zabezpečení ochrany dat. Bez ohledu na způsob, který hodláte používat, musíte zajistit, že budou splněny některé nezbytné předpoklady:

- Musíte splnit nezbytné předpoklady pro instalaci a použití produktu DCM (Digital Certificate Manager).
- Musíte vytvořit paměť certifikátů \*SIGNATUREVERIFICATION. Tuto paměť certifikátů můžete vytvořit jedním ze dvou způsobů, záleží pouze na vašich potřebách. Můžete ji vytvořit pomocí produktu DCM (Digital Certificate Manager) při správě vašich certifikátů pro ověřování podpisů. Anebo, pokud používáte k podepisování objektů veřejný certifikát, můžete vytvořit tuto paměť certifikátů pomocí vámi napsaného programu, který bude používat rozhraní Add Verifier (QYDOADDV) API.

**Poznámka:** Rozhraní Add Verifier API vytvoří paměť certifikátů s předvoleným heslem. Musíte pomocí produktu DCM přenastavit toto předvolené heslo na jiné (podle vaší volby), abyste zabránili neoprávněnému přístupu do paměti certifikátů.

- Paměť certifikátů \*SIGNATUREVERIFICATION musí obsahovat kopii certifikátu, kterým jsou objekty podepsané. Tento certifikát můžete přidat do paměti certifikátů dvěma způsoby. Můžete použít produkt DCM na podepisujícím systému a vyexportovat certifikát do souboru a potom pomocí produktu DCM v cílovém ověřovacím systému naimportovat certifikát do paměti certifikátů \*SIGNATUREVERIFICATION. Anebo, pokud používáte k podepisování objektů veřejný certifikát, můžete přidat certifikát do paměti certifikátů v cílovém ověřovacím systému prostřednictvím vámi napsaného programu, který používá rozhraní Add Verifier API.

- Paměť certifikátů \*SIGNATUREVERIFICATION musí obsahovat kopii certifikátu CA, a to kopii certifikátu, kterým jsou objekty podepsané. Pokud používáte k podepisování objektů veřejný certifikát, paměť certifikátů v cílovém ověřovacím systému by měla mít také kopii požadovaného certifikátu CA. Pokud však k podepisování objektů používáte certifikát vydaný lokálním vydavatelem certifikátů (CA), musíte použít produkt DCM k přidání kopie certifikátu lokálního vydavatele certifikátů (CA) do paměti certifikátů v cílovém ověřovacím systému.

**Poznámka:** Z bezpečnostních důvodů rozhraní Add Verifier API neumožňuje vložit certifikát vydavatele certifikátů (CA) do paměti certifikátů \*SIGNATUREVERIFICATION. Když přidáte certifikát CA do paměti certifikátů, systém předpokládá, že CA je ověřený zdroj certifikátů. Následkem toho systém zachází s certifikátem vydaným CA, jako kdyby byl vydán ověřeným zdrojem. Z tohoto důvodu nemůžete používat rozhraní API k vytvoření programu uživatelského vstupu, který by vložil certifikát CA do paměti certifikátů. K přidání certifikátu CA do paměti certifikátů musíte použít produkt DCM (Digital Certificate Manager), abyste zajistili, že někdo musí speciálně a ručně řídit, kterým vydavatelům certifikátu (CA) systém důvěřuje. Tímto způsobem předejete možným případům, kdy systém mohl importovat certifikáty ze zdrojů, které administrátor vědomě nezadal jako ověřené.

Pokud k podepisování objektů používáte certifikát vydaný lokálním vydavatelem certifikátů (CA), musíte pomocí produktu DCM na hostitelském serveru iSeries lokálního vydavatele certifikátů (CA) vyexportovat kopii certifikátu lokálního CA do souboru. Pak můžete pomocí produktu DCM na cílovém ověřovacím serveru iSeries naimportovat certifikát lokálního CA do paměti certifikátů \*SIGNATUREVERIFICATION. Chcete-li předejít možným chybám, musíte naimportovat certifikát lokálního CA do paměti certifikátů dříve, než použijete rozhraní Add Verifier API k přidání certifikátu pro ověřování podpisů. Pokud tedy používáte certifikát vydaný lokálním CA, můžete snadnějším způsobem použít produkt DCM k naimportování jak certifikátu lokálního CA, tak i certifikátu pro ověřování podpisů do paměti certifikátů.

Pokud chcete komukoliv zabránit v používání tohoto rozhraní API k přidávání ověřovacího certifikátu do vaší paměti certifikátů \*SIGNATUREVERIFICATION bez vašeho vědomí, měli byste uvažovat o zablokování daného rozhraní API ve vašem systému. Tento úkon můžete provést pomocí nástrojů SST (system service tools), které zamítnou změny v systémových hodnotách týkajících se zabezpečení. .

- Uživatelský profil iSeries, který ověřuje podpisy, musí mít speciální oprávnění \*AUDIT. Uživatelský profil iSeries, který vytváří paměť certifikátů \*SIGNATUREVERIFICATION nebo mění její heslo, musí mít speciální oprávnění \*SECADM a \*ALLOBJ.

---

## Správa podepsaných objektů

Počínaje verzí V5R1 začala společnost IBM podepisovat licencované programy OS/400 a PTF, jako způsob oficiálního označení toho, že operační systém pochází od IBM, a jako prostředek detekce neoprávněných změn, které by se mohly objevit na systémových objektech. Také obchodní partneři a ostatní prodejci mohou podepisovat aplikace, které můžete zakoupit. Z těchto důvodů i přesto, že nechcete sami podepisovat objekty, potřebujete rozumět tomu, jak pracovat s podepsanými objekty a jak tyto podepsané objekty ovlivňují úlohy týkající se administrativy systému.

Podepsané objekty primárně ovlivňují úlohy zálohování a obnovy dat, konkrétně způsob, jakým ukládáte a obnovujete objekty ve vašem systému.

### Systémové hodnoty a příkazy, které ovlivňují podepsané objekty

Zjistěte, jaké systémové hodnoty a příkazy můžete používat při správě podepsaných objektů, a které z nich mají vliv na podepsané objekty, jestliže je spustíte.

### **Pokyny pro ukládání a obnovu podepsaných objektů**

Zjistěte, jaký vliv mají podepsané objekty na způsob, jakým provádíte úlohy ukládání a obnovy ve vašem systému.

### **Příkazy programu pro kontrolu kódu k zajištění integrity podpisu**

Zjistěte podrobnosti o tom, jak používat příkazy pro ověřování podpisů objektů, abyste stanovili integritu objektu.

## **Systemové hodnoty a příkazy, které ovlivňují podepsané objekty**

Chcete-li efektivně spravovat podepsané objekty, musíte porozumět tomu, jak systémové hodnoty a příkazy ovlivňují podepsané objekty. Systémová hodnota **QVFYOBJRST (Verify object signatures during restore)** určuje, jak určité příkazy pro obnovu ovlivňují podepsané objekty a jak bude váš systém zacházet s podepsanými objekty během operací obnovy dat. Neexistují žádné CL příkazy, které by byly navrženy výhradně pro práci s podepsanými objekty v systémech iSeries. Existuje však velký počet běžných CL příkazů, které můžete používat ke správě podepsaných objektů (nebo ke správě objektů infrastruktury, které učiní podepisování objektů možným). Jiné příkazy mohou negativně ovlivnit podepsané objekty ve vašem systému např. tím, že odstraní podpisy z objektů, čímž zruší ochranu, kterou poskytuje podpis.

### **Systemové hodnoty, které ovlivňují podepsané objekty**

Systémová hodnota **QVFYOBJRST (Verify object signatures during restore)**, člen kategorie systémových hodnot operačního systému OS/400 vztahujících se k obnově dat, určuje, jak příkaz ovlivňuje podepsané objekty ve vašem systému. Tato systémová hodnota, kterou je možné ovládat prostřednictvím produktu iSeries Navigator, řídí, jak bude systém pracovat s ověřováním podpisů během operací obnovy dat. Nastavení, které použijete pro tuto systémovou hodnotu, v kombinaci s nastaveními dalších dvou systémových hodnot ovlivní operace obnovy dat ve vašem systému. V závislosti na nastavení, které jste pro tuto hodnotu zvolili, můžete povolit nebo nepovolit obnovu objektů na základě jejich stavu podpisu. (Například je-li objekt nepodepsán, má neplatný podpis, je podepsán ověřeným zdrojem, atd.) Předvolené nastavení pro tuto systémovou hodnotu umožňuje, aby nepodepsané objekty byly obnoveny, ale zajišťuje, že podepsané objekty lze obnovit pouze tehdy, mají-li objekty platný podpis. Systém definuje objekt jako podepsaný pouze v tom případě, že objekt má podpis, kterému váš systém důvěřuje. Jiné, "nedůvěryhodné", podpisy na objektu systém ignoruje a pracuje s objektem, jako kdyby byl nepodepsaný.

Pro systémovou hodnotu QVFYOBJRST lze nastavit několik hodnot, od ignorování všech podpisů, až po vyžadování platných podpisů u všech objektů, které systém obnovuje. Tato systémová hodnota ovlivňuje pouze spustitelné objekty, které jsou obnovovány, jako jsou programy (\*PGM), příkazy (\*CMD), servisní programy (\*SRVPGM), SQL balíky (\*SQLPKG) a moduly (\*MODULE). Používá se také pro objekty typu proudový soubor (\*STMF), které mají přidruženy Java programy, vytvořené příkazem CRTJVAPGM. Nepoužívá se pro soubory typu save (\*SAV) nebo IFS soubory.

Další informace o používání této a dalších systémových hodnot uvádí téma System Value Finder v rámci aplikace Information Center.

### **CL příkazy, které ovlivňují podepsané objekty**

Existuje několik CL příkazů, které vám dovolují pracovat s podepsanými objekty nebo které ovlivňují podepsané objekty na vašem serveru iSeries. Můžete používat celou škálu příkazů k tomu, abyste si prohlíželi informace o podpisu objektů, ověřovali podpis na objektech a ukládali a obnovovali objekty zabezpečení ochrany dat nutné pro ověřování podpisů. Navíc zde existuje skupina příkazů, které, jsou-li spuštěny, mohou odstranit podpis z objektů a popřít tak zabezpečení ochrany dat, které poskytuje používání podpisů.

### **Příkazy pro prohlížení informace o podpisu na objektu**

- Příkaz DSPOBJD (Display Object Description).  
Tento příkaz zobrazí jména a atributy zadaných objektů v uvedené knihovně nebo v knihovnách seznamu knihoven vlákna. Tento příkaz můžete používat k určení, zda je objekt podepsán a k zobrazení informací o podpisu.
- Příkazy integrovaného systému souborů DSPLNK (Display Object Links) a WRKLNK (Work with Object Links).  
Kterýkoliv z těchto příkazů můžete použít k zobrazení informací o podpisu na objektu v integrovaném systému souborů.

### **Příkazy pro ověřování podpisů na objektech**

- Příkaz CHKOBJTG (Check Object Integrity).  
Tento příkaz dovoluje určit, zda mají objekty ve vašem systému narušenu integritu. Tímto příkazem můžete ověřovat podpisy stejným způsobem, jako používáte program pro vyhledávání virů k určení, zda virus narušil soubory nebo jiné objekty ve vašem systému. Chcete-li se dozvědět více o používání tohoto příkazu pro práci s podepsanými a podepisovatelnými objekty, prostudujte si část Příkazy programu pro kontrolu kódu k zajištění integrity podpisu.
- Příkaz CHKPRDOPT (Check Product Option).  
Tento příkaz hlásí rozdíly mezi správnou a aktuální strukturou softwarového produktu. Příkaz například hlásí chybu, pokud je z instalovaného produktu odstraněn nějaký objekt. Pomocí parametru CHKSIG můžete zadat, jak má příkaz zpracovat a hlásit možné problémy s podpisem u produktu. Chcete-li se dozvědět více o používání tohoto příkazu pro práci s podepsanými a podepisovatelnými objekty, prostudujte si část Příkazy programu pro kontrolu kódu k zajištění integrity podpisu.
- Příkaz SAVLICPGM (Save Licensed Program).  
Tento příkaz ukládá kopii objektů, které vytváří licenční program. Ukládá licenční program ve formě, ze které je možné jej obnovit pomocí příkazu RSTLICPGM (Restore Licensed Program). Pomocí parametru CHKSIG můžete zadat, jak má příkaz zpracovat a hlásit možné problémy s podpisem u produktu. Chcete-li se dozvědět více o používání tohoto příkazu pro práci s podepsanými a podepisovatelnými objekty, prostudujte si část Příkazy programu pro kontrolu kódu k zajištění integrity podpisu.
- Příkaz RST (Restore).  
Tento příkaz obnoví kopii jednoho nebo více objektů, které je možné používat v integrovaném systému souborů (IFS). Tento příkaz dále dovoluje obnovit paměti certifikátů a jejich obsahy v systému. Nemůžete však používat tento příkaz k obnovení paměti certifikátů \*SIGNATUREVERIFICATION. To, jak příkaz pro obnovu zpracovává podepsané a podepisovatelné objekty, je určeno nastavením systémové hodnoty QVFYOBJRST (Verify object signatures during restore).
- Příkaz RSTLIB (Restore Library).  
Tento příkaz obnoví jednu knihovnu nebo skupinu knihoven, které byly uloženy příkazem SAVLIB (Save Library). Příkaz RSTLIB obnoví celou knihovnu, včetně popisu knihovny, popisů objektů a obsahů objektů v knihovně. To, jak tento příkaz zpracovává podepsané a podepisovatelné objekty, je určeno nastavením systémové hodnoty QVFYOBJRST (Verify object signatures during restore).
- Příkaz RSTLICPGM (Restore Licensed Program).  
Tento příkaz zavádí nebo obnovuje licenční program, buď pro počáteční instalaci, nebo pro instalaci nového vydání. To, jak tento příkaz zpracovává podepsané a podepisovatelné objekty, je určeno nastavením systémové hodnoty QVFYOBJRST (Verify object signatures during restore).
- Příkaz RSTOBJ (Restore Object).  
Tento příkaz obnoví jeden nebo více objektů v jediné knihovně, které byly uloženy na disketu, pásku, optický nosič nebo do souboru typu save pomocí jediného příkazu. To, jak tento příkaz zpracovává podepsané a podepisovatelné objekty, je určeno nastavením systémové hodnoty QVFYOBJRST (Verify object signatures during restore).

### **Příkazy pro ukládání a obnovu paměti certifikátů**

- Příkaz SAV (Save).  
Tento příkaz vám dovolí uložit kopii jednoho nebo více objektů, které je možné používat v integrovaném systému souborů, včetně paměti certifikátů. Nemůžete však používat tento příkaz k uložení paměti certifikátů \*SIGNATUREVERIFICATION.

- Příkaz SAVSECDTA (Save Security Data).  
Tento příkaz dovoluje uložit všechny informace o zabezpečení ochrany dat bez požadavku, aby byl systém ve stavu omezení. Prostřednictvím tohoto příkazu můžete uložit paměť certifikátů \*SIGNATUREVERIFICATION a certifikáty, které tato paměť certifikátů obsahuje. Tento příkaz neuloží žádnou jinou paměť certifikátů.
- Příkaz SAVSYS (Save System).  
Tento příkaz dovoluje uložit kopii interního kódu LIC a knihovnu QSYS ve formátu kompatibilním s instalací serveru iSeries. Neukládá žádné jiné objekty z žádné jiné knihovny. Kromě toho dovoluje uložit konfigurační objekty a objekty zabezpečení ochrany, které můžete také uložit pomocí příkazů SAVSECDTA a SAVCFG. Pomocí tohoto příkazu můžete uložit paměť certifikátů \*SIGNATUREVERIFICATION a certifikáty, které obsahuje.
- Příkaz RST (Restore).  
Tento příkaz dovoluje obnovit paměti certifikátů a jejich obsah v systému. Tento příkaz však nemůžete používat k obnově paměti certifikátů \*SIGNATUREVERIFICATION.
- Příkaz RSTUSRPRF (Restore User Profiles).  
Tento příkaz dovoluje obnovit základní části uživatelského profilu nebo sady uživatelských profilů, uložených příkazem SAVSYS (Save System) nebo SAVSECDTA (Save Security Data). Tento příkaz můžete použít k obnově paměti certifikátů \*SIGNATUREVERIFICATION a uložených hesel této paměti certifikátů a všech ostatních pamětí certifikátů. Paměť certifikátů \*SIGNATUREVERIFICATION můžete obnovit aniž byste obnovili informace o uživatelském profilu, pokud zadáte \*DCM do parametru SECDTA a \*NONE do parametru USRPRF. Chcete-li tímto příkazem obnovit informace o uživatelském profilu a paměti certifikátů, včetně jejich hesel, zadejte \*ALL do parametru USRPRF.

### Příkazy, které mohou odstranit nebo ztratit podpisy z objektů

Pokud aplikujete některý z následujících příkazů na podepsaný objekt, můžete to udělat takovým způsobem, kterým byste mohli vymazat nebo ztratit podpis z objektu. Odstranění podpisu by mohlo způsobit problémy s takto ovlivněným objektem. Minimálně byste nebyli nadále schopni ověřovat zdroj objektu, zda je důvěryhodný, a nebyli byste schopni ani ověřovat podpisy, abyste zjistili, zda na objektu byly provedeny nějaké změny. Tyto příkazy byste měli používat pouze na ty podepsané objekty, které jste vytvořili (na rozdíl od podepsaných objektů, které jste obdrželi zvenčí, jako např. od IBM nebo jiných dodavatelů). Pokud máte obavy, že příkaz odstraní nebo ztratí podpis objektu, můžete pomocí příkazu DSPOBJD (Display Object Description) zjistit, zda je na objektu stále podpis a, je-li to nezbytné, objekt znovu podepsat.

**Poznámka:** Chcete-li ověřit, zda během provádění příkazu pro uložení došlo ke ztrátě podpisu objektu, musíte tento objekt obnovit do jiné knihovny, než ve které jste jej uložili (například do QTEMP). Pak můžete použít příkaz DSPOBJD, abyste zjistili, zda objekt na záložním médiu ztratil svůj podpis.

- Příkaz CHGPGM (Change Program).  
Tento příkaz změní atributy programu, aniž by vyžadoval jeho opětovnou kompilaci. Tento příkaz můžete také použít k vynucení opětovného vytvoření programu, i když byly uvedeny stejné atributy, jako jsou aktuální atributy.
- Příkaz CHGSRVPGM (Change Service Program).  
Tento příkaz změní atributy servisního programu, aniž by vyžadoval jeho opětovnou kompilaci. Tento příkaz můžete také použít k vynucení opětovného vytvoření servisního programu, i když byly uvedeny stejné atributy, jako jsou aktuální atributy.
- Příkaz CLRSAVF (Clear Save File).  
Tento příkaz vymaže obsah souboru typu save. Vymaže všechny stávající záznamy ze souboru typu save a sníží množství paměti, které tento soubor používá.
- Příkaz SAV (Save).  
Tento příkaz uloží kopii jednoho nebo více objektů, které je možné používat v integrovaném systému souborů. Pokud používáte tento příkaz, můžete ztratit podpis z objektů \*CMD, uložených na záložních



médiích, jestliže zadáte hodnotu parametru TGTRLS nižší než V5R2M0. Ztráta podpisu se objeví, protože objekty \*CMD nebylo možné ve verzích před verzí V5R2 podepisovat.

- Příkaz SAVLIB (Save Library).  
Tento příkaz dovoluje uložit kopii jedné nebo více knihoven. Pokud používáte tento příkaz, můžete ztratit podpis z objektů \*CMD uložených na záložních médiích, pokud zadáte hodnotu parametru TGTRLS nižší než V5R2M0. Ztráta podpisu se objeví, protože objekty \*CMD nebylo možné ve verzích před verzí V5R2 podepisovat.
- Příkaz SAVOBJ (Save Object).  
Tento příkaz uloží kopii jednoho objektu nebo skupiny objektů, umístěných ve stejné knihovně. Pokud používáte tento příkaz, můžete ztratit podpis z objektů \*CMD, uložených na záložních médiích, pokud zadáte hodnotu parametru TGTRLS nižší než V5R2M0. Ztráta podpisu se objeví, protože objekty \*CMD nebylo možné ve verzích před verzí V5R2 podepisovat.

## Pokyny pro ukládání a obnovu podepsaných objektů

Několik systémových hodnot může ovlivnit operace obnovy vašeho serveru iSeries. Pouze jedna z těchto systémových hodnot, systémová hodnota **QVfyOjRST (Verify object signatures during restore)**, určuje, jak bude systém pracovat s podepsanými objekty během jejich obnovy. Nastavení, pro které se rozhodnete u této systémové hodnoty, vám umožňuje určit, jak bude proces obnovy pracovat s ověřováním objektů bez podpisů nebo objektů s neplatnými podpisy.

Některé příkazy pro ukládání a obnovu dat ovlivňují podepsané objekty nebo určují, jak bude váš systém pracovat s podepsanými a nepodepsanými objekty během operací ukládání a obnovy dat. Měli byste si být vědomi možností a vlivu těchto příkazů na podepsané objekty, abyste mohli lépe spravovat váš systém a abyste se vyvarovali možných problémů, které by mohly nastat.

Níže uvedené příkazy mohou ověřovat podpisy na objektech během operací uložení a obnovy dat:

- Příkaz SAVLICPGM (Save Licensed Program).
- Příkaz RST (Restore).
- Příkaz RSTLIB (Restore Library).
- Příkaz RSTLICPGM (Restore Licensed Program).
- Příkaz RSTOBJ (Restore Object).

Níže uvedené příkazy dovolují uložit a obnovit paměti certifikátů. Paměti certifikátů jsou objekty citlivé na utajení, které obsahují certifikáty, které používáte k podepisování objektů a ověřování podpisů. Jde o tyto příkazy:

- Příkaz SAV (Save).
- Příkaz SAVSECDTA (Save Security Data).
- Příkaz SAVSYS (Save System).
- Příkaz RST (Restore).
- Příkaz RSTUSRPRF (Restore User Profiles).

Některé příkazy pro uložení mohou v určitých případech (v závislosti na použitých hodnotách parametrů) ztratit podpis z objektu uloženého na záložním médiu, čímž zruší ochranu, kterou poskytuje podpis. Například *jakákoliv* operace uložení, která se odkazuje na objekt \*CMD s vydáním v cílovém systému dřívějším než V5R2M0, způsobí, že takový příkaz bude uložen bez podpisu. Odstranění podpisu by mohlo způsobit problémy s takto ovlivněným objektem. Minimálně byste nebyli nadále schopni ověřovat zdroj objektu, zda je důvěryhodný, a nebyli byste schopni ani ověřovat podpisy, abyste zjistili, zda na objektu byly provedeny nějaké změny. Tyto příkazy byste měli používat pouze na ty podepsané objekty, které jste vytvořili (na rozdíl od podepsaných objektů, které jste obdrželi zvenčí, jako např. od IBM nebo jiných dodavatelů).

**Poznámka:** Chcete-li ověřit, zda příkaz pro uložení ztratil podpis objektu, musíte tento objekt obnovit do jiné knihovny, než ve které jste jej uložili (například do QTEMP). Pak můžete použít příkaz DSPOBJD, abyste zjistili, zda objekt na záložním médiu ztratil svůj podpis.

Měli byste si být vědomi těchto schopností níže uvedených specifických příkazů uložení, stejně tak jako příkazů uložení obecně:

- Příkaz SAV (Save).
- Příkaz SAVLIB (Save Library).
- Příkaz SAVOBJ (Save Object).

Další informace o tom, jak tyto příkazy ovlivňují podepsané objekty a podpisy objektů během operací uložení a obnovy dat, prostudujte si část Systémové hodnoty a příkazy, které ovlivňují podepsané objekty.

## Příkazy programu pro kontrolu kódu k zajištění integrity podpisu

Pomocí produktu DCM (Digital Certificate Manager) nebo pomocí rozhraní API můžete ověřovat podpisy na objektech. Ke kontrole podpisů můžete také používat několik příkazů. Použití těchto příkazů dovoluje ověřovat podpisy v podstatě stejným způsobem, jako používáte program pro vyhledávání virů k určení, zda virus narušil soubory nebo jiné objekty ve vašem systému. Většina podpisů se kontroluje během obnovy objektu nebo jeho instalace na serveru, například při používání příkazu RSTLIB.

Můžete si vybrat jeden ze tří příkazů, pomocí kterých je možné zkontrolovat podpisy na objektech, které jsou již v systému. Jeden z těchto příkazů, příkaz CHKOBJITG (Check Object Integrity), je navržen speciálně pro ověřování podpisů objektů. Kontrola podpisů u každého z těchto příkazů je řízena parametrem CHKSIG. Uvedený parametr dovoluje kontrolovat podpisy všech typů objektů, které je možné podepsat, ignorovat všechny podpisy, nebo kontrolovat pouze objekty, které mají podpisy. Tato poslední volba je nastavena jako předvolená hodnota uvedeného parametru.

### Příkaz CHKOBJITG (Check Object Integrity)

Příkaz CHKOBJITG (Check Object Integrity) dovoluje určit, zda mají objekty ve vašem systému narušenu integritu. Tímto příkazem můžete zkontrolovat narušení integrity objektů, které jsou vlastněny určitým uživatelským profilem, objektů, které odpovídají určitému jménu cesty, nebo všech objektů v systému. Záznam protokolu o narušení integrity se objeví, je-li splněna jedna z těchto podmínek:

- Objekt příkazu, programu nebo modulu, nebo atributy knihovny byly pozměněny.
- Digitální podpis na objektu byl určen jako neplatný. Podpis je zašifrovaný matematický součet dat v objektu. Z tohoto důvodu se považuje, že podpis odpovídá a je platný, pokud data v objektu během ověřování odpovídají datům v objektu ve chvíli, kdy byl podepisován. Neplatný podpis se určuje na základě porovnání zašifrovaného matematického součtu, jenž byl vytvořen při podpisu objektu, a zašifrovaného matematického součtu vytvořeného během ověřování podpisu. Proces ověřování podpisů porovná obě součtové hodnoty. Pokud hodnoty nejsou stejné, obsah objektu byl od okamžiku jeho podpisu změněn a podpis se považuje za neplatný.
- Objekt má chybný atribut domény pro typ objektu.
- 

Pokud příkaz zaznamená narušení integrity objektu, přidá jméno objektu, jméno knihovny (nebo jméno cesty), typ objektu, vlastníka objektu a typ selhání do databázového souboru protokolu. Příkaz také vytvoří záznam protokolu v některých dalších případech, přestože v těchto případech nejde o narušení integrity. Příkaz například vytvoří záznam do protokolu pro objekt, který je podepisovatelný, ale který nemá digitální podpis, dále vytvoří záznam pro objekty, které nebylo možné zkontrolovat, a také pro objekty ve formátu, jenž vyžaduje změny, aby je bylo možné použít v aktuální implementaci systému (konverze IMPI na RISC).

Hodnota parametru CHKSIG řídí, jak bude příkaz pracovat s digitálními podpisy na objektech. Můžete zadat jednu z následujících tří hodnot:

- \*SIGNED – Zadáte-li tuto hodnotu, příkaz zkontroluje objekty s digitálními podpisy. Příkaz vytvoří záznam do protokolu pro každý objekt, jehož podpis není platný. Toto je předvolená hodnota.
- \*ALL – Zadáte-li tuto hodnotu, příkaz zkontroluje všechny podepisovatelné objekty, aby určil, zda mají podpis. Příkaz vytvoří záznam do protokolu pro každý podepisovatelný objekt, který nemá podpis, nebo pro každý objekt, jehož podpis není platný.
- \*NONE – Zadáte-li tuto hodnotu, příkaz nezkontroluje digitální podpisy na objektech.

#### **Příkaz CHKPRDOPT (Check Product Option)**

Příkaz CHKPRDOPT (Check Product Option) hlásí rozdíly mezi správnou a aktuální strukturou softwarového produktu. Příkaz například hlásí chybu, pokud je z instalovaného produktu objekt odstraněn.

Hodnota parametru CHKSIG řídí, jak bude příkaz pracovat s digitálními podpisy na objektech. Můžete zadat jednu z následujících tří hodnot:

- \*SIGNED – Zadáte-li tuto hodnotu, příkaz zkontroluje objekty s digitálními podpisy. Příkaz zkontroluje podpisy na libovolném podepsaném objektu. Pokud příkaz určí, že podpis na objektu není platný, příkaz odešle zprávu do protokolu úlohy a označí, že je produkt v chybném stavu. Toto je předvolená hodnota.
- \*ALL – Zadáte-li tuto hodnotu, příkaz zkontroluje všechny podepisovatelné objekty, aby určil, zda mají podpis a ověřil podpisy na těchto objektech. Příkaz odešle zprávu do protokolu úlohy pro každý podepisovatelný objekt, který nemá podpis. Příkaz však neoznačí produkt jako chybný. Pokud příkaz určí, že podpis na objektu není platný, odešle zprávu do protokolu úlohy a označí produkt jako chybný.
- \*NONE – Zadáte-li tuto hodnotu, příkaz nezkontroluje digitální podpisy na objektech.

#### **Příkaz SAVLICPGM (Save Licensed Program)**

Příkaz SAVLICPGM (Save Licensed Program) dovoluje ukládat kopie objektů, které vytváří licenční program. Ukládá licenční program ve formě, ze které je možné jej obnovit pomocí příkazu RSTLICPGM (Restore Licensed Program).

Hodnota parametru CHKSIG řídí, jak bude příkaz pracovat s digitálními podpisy na objektech. Můžete zadat jednu z následujících tří hodnot:

- \*SIGNED – Zadáte-li tuto hodnotu, příkaz zkontroluje objekty s digitálními podpisy. Příkaz zkontroluje podpisy na libovolném podepsaném objektu, ale nezkontroluje nepodepsané objekty. Pokud příkaz určí, že podpis na objektu není platný, příkaz odešle zprávu do protokolu úlohy, aby označil objekt, a operace uložení skončí s chybou. Toto je předvolená hodnota.
- \*ALL – Zadáte-li tuto hodnotu, příkaz zkontroluje všechny podepisovatelné objekty, aby určil, zda mají podpis, a ověřil podpisy na těchto objektech. Příkaz odešle zprávu do protokolu úlohy pro každý podepisovatelný objekt, který nemá podpis. Proces ukládání však nebude ukončen. Pokud příkaz určí, že podpis na objektu není platný, příkaz odešle zprávu do protokolu úlohy a proces ukládání bude ukončen s chybou.
- \*NONE – Zadáte-li tuto hodnotu, příkaz nezkontroluje digitální podpisy na produktových objektech.

---

## **Odstraňování problémů s podepsanými objekty**

V následující tabulce naleznete informace, které vám pomohou při odstraňování některých běžnějších problémů, se kterými se můžete setkat při práci se schopnostmi serverů iSeries v oblasti podepisování dat a ověřování podpisů.

### **Běžné problémy s podepisováním objektů**



Problém	Možné řešení
Při použití rozhraní Sign Object API k podepsání objektu s cílovým vydáním V4R5 (nebo nižším) proces podepisování selže a objekt není podepsán (chybová zpráva CPF721).	iSeries neposkytuje podporu pro podepisování objektů až do V5R1. Pro takové objekty, které vracejí chybovou zprávu CPF721, musíte znovu vytvořit programy, aby měly v cílovém systému vydání V5R1 nebo vyšší, aby bylo možné je podepsat.

### Běžné problémy ověřování podpisů

Problém	Možné řešení
Pro objekty bez podpisů selhal proces obnovy.	Pokud nezbytnost podpisu není středem zájmu, zkontrolujte, zda systémová hodnota QVFOBJRST je nastavena na 5. Hodnota 5 udává, že podepsané objekty nelze obnovit. Změňte tuto hodnotu na 3 a zkuste obnovu znovu.
Pro objekty s podpisy selhal proces obnovy.	Toto se může stát, pokud paměť certifikátů *SIGNATUREVERIFICATION byla přenesena do systému, ale nebyl použit produkt DCM ke změně jejího hesla. V takovém případě není možné použít certifikáty, které paměť certifikátů obsahuje, k ověření podpisů na objektech během procesu obnovy. Použijte produkt DCM, abyste změnili heslo paměti certifikátů. Pokud neznáte heslo, musíte vymazat paměť certifikátů, znovu ji vytvořit a pomocí produktu DCM změnit heslo.
Během obnovy nebo instalace produktu se objevuje chyba, protože selhalo ověření podpisu.	Pokud nelze ověřit, že podpis objektu je správný, může chyba znamenat, že objekt byl od svého podpisu změněn. Pokud je integrita objektu základním měřítkem, neměli byste změnit systémovou hodnotu QVFOBJRST nebo provádět jiné akce, které by mohly umožnit, aby se podezřelý objekt obnovil. Takovým způsobem byste mohli obejít zabezpečení ochrany dat, které poskytuje ověřování podpisů, a dovolili byste, aby se škodlivý objekt objevil ve vašem systému. Namísto toho byste se měli obrátit na signatáře objektu, aby určil odpovídající akce, které by se měly provést pro vyřešení problému.

## Související informace pro podepisování objektů a ověření podpisů

Podepisování objektů a ověření podpisů jsou relativně nové technologie zabezpečení ochrany dat. Uvádíme krátký seznam dalších zdrojů, které by mohly být pro vás užitečné, pokud se zajímáte o širší porozumění těmto technologiím a mechanismům, jak pracují:

- **Webové stránky VeriSign Help Desk**  Webové stránky VeriSign poskytují rozsáhlou knihovnu, věnovanou problematice digitálních certifikátů, jako např. podepisování objektů, ale i řadě dalších témat týkajících se bezpečnosti Internetu.
- **IBM eServer iSeries Wired Network Security: OS/400 V5R1 DCM and Cryptographic Enhancements SG24-6168**  Tato červená kniha IBM je zaměřena na možnosti zlepšení zabezpečení sítě ve verzi V5R1. Pokrývá mnoho témat, např. jak využívat schopností serveru iSeries v oblasti podepisování objektů, jak používat produkt DCM (Digital Certificate Manager), atd.





Vytištěno v Dánsku společností IBM Danmark A/S.