Jednotný standard pro správu klíčů

Heterogenní charakter dnešního světa ICT technologie činí správu kryptografických klíčů složitější než dříve. Standardů je sice řada, ale univerzální podnikový standard nezávislý na výrobci byl ještě před dvěma roky v nedohlednu. Dnes je vše jinak.

Na všeobecně použitelný standard v oblasti správy klíčů čeká bezpečnostní komunita již mnoho let. K dispozici je řada standardů, ale jsou určeny převážně pro specifické použití (finanční instituce, státní organizace, platební karty, autentizační tokeny, mobilní telefony… blíže [4]). Mezinárodní standard ISO/IEC 11770 zase není (viz [9]) podnikovou sférou všeobecně přijímán, snad že se vůbec nezabývá životním cyklem klíčů.

V únoru roku 2009 společnosti Brocade, EMC/RSA, HP, IBM, LSI, NetApp, Seagate a Thales e-Security iniciovaly v rámci organizace OASIS[1] (Organization for the Advancement of Structured Information Standards) vznik technického výboru KMIP (Key Management Interoperability Protocol). Postupně se připojila řada dalších významných ICT společností[2].

Loni v listopadu poskytli autoři odborné veřejnosti (potenciální uživatelé, vývojáři atd.) aktuální verzi specifikace k připomínkám. Ty pak zahrnuli do další verze draftu a opravené materiály dali veřejnosti opět k dispozici letos v dubnu.

Na březnové konferenci RSA v USA proběhla v rámci OASIS Interop Showcase demonstrace praktického použití zkušební verze 1.0 KMIP. Serverové aplikace předváděly společnosti HP a IBM, klientské navíc SafeNet. Do října má být verze 1.0 organizací OASIS schválena pod označením 1 jako první oficiální verze protokolu.

Jak to, že se za rok a půl daří vydat nový standard? Je to jedině díky nebývalé podpoře předních výrobců ICT, kteří do vývoje zapojili své nejlepší odborníky z oblasti kryptografie. Úzké firemní zájmy šly stranou, o čemž svědčí i to, že práce koordinovali Robert Haas z IBM a Indra Fitzgerald z HP, neboli z dvou jinak si tvrdě konkurujících firem.

Přitom OASIS již čtyři roky ne a ne dokončit jiný standard[3] připravovaný v rámci pracovního výboru EKMI (Enterprise Key Management Infrastructure). Těžko říci, zda velké ICT společnosti přesunuly svůj zájem z tohoto protokolu na KMIP pro dosavadní vleklost prací, či naopak je zpoždění prací na EKMI důsledkem přesunu pozornosti firem k KMIP.

Protokol KMIP řeší aktuální problém nedostatečné interoperability kryptografických zařízení v konfiguraci klient – server, proto je mu věnován tento článek. Seznámení s ním proběhne v pořadí: charakteristika protokolu, specifikace, zprávy, použité kryptografické funkce. V závěru si zhodnotíme, co lze od protokolu dále očekávat.

Charakteristika protokolu

Cílem protokolu KMIP je vytvoření jednotného a zároveň komplexního prostředí pro komunikaci mezi servery správy šifrovacích klíčů a jejich klienty. Protokol by měl být použitelný na jakoukoli šifrovací aplikaci zákazníka – od mobilního zařízení po diskové pole, resp. páskovou paměť libovolného výrobce.

Díky podpoře KMIP budou moci podniky soustředit správu klíčů na jednom místě, snížit operační náklady na klíčovou infrastrukturu a zároveň posílit kontrolu dodržování bezpečnostní politiky. Za unifikaci se platí odhlédnutím od některých specifických požadavků. Je to nezbytné, protože např. požadavky

[1] Nevýdělečné konsorcium, které řídí vývoj, konvergenci a adaptaci otevřených standardů pro globální informační komunitu.

[2] Algorithmic Research, Axway Software, BeCrypt, CipherOptics, Dajeil, Election Systems and Software, Emulex, Lexmark International, MIT, Mitre Corporation, NIST, Oracle, PayPal, PGP Corporation, Quantum, Red Hat, SafeNet, Skyworth TTG, Sun Microsystems, Symantec, Valicore, Venafi, Verisign a další. K podpoře protokolu se přihlásilo i Ministerstvo obrany USA.

[3] Má zajistit správu symetrických klíčů s využitím jazyka XML.

```
Tag: Request Message (0x420078), Type: Structure (0x01), Data:
   Tag: Request Header (0x420077), Type: Structure (0x01), Data:
      Tag: Protocol Version (0x420069), Type: Structure (0x01), Data:
         Tag: Protocol Version Major (0x42006A), Type: Integer (0x02), Data: 0x00000001 (1)
         Tag: Protocol Version Minor (0x42006B), Type: Integer (0x02), Data: 0x00000000 (0)
      Tag: Batch Count (0x42000D), Type: Integer (0x02), Data: 0x00000001 (1)
   Tag: Batch Item (0x42000F), Type: Structure (0x01), Data:
      Tag: Operation (0x42005C), Type: Enumeration (0x05), Data: 0x0000000A (Get)
      Tag: Request Payload (0x420079), Type: Structure (0x01), Data:
         Tag: Unique Identifier (0x420094), Type: Text String (0x07), Data: 0a33e83e-5b7a-
         4865-964a-8d1c3bbf9ae3
```

Obr. 1: Příklad kódování zprávy – zápis zprávy Get(symmetric key).

na způsob ochrany dat disku notebooku jsou odlišné od požadavků na ochranu disku ve výpočetním centru.

Technický výbor publikoval normativní specifikaci protokolu [5], návod, jak protokol implementovat [6], příklady uživatelských aplikací [7] a popis kryptografických profilů [8].

Specifikace protokolu

Specifikace protokolu má řešit očekávané požadavky zákazníků na správu životního cyklu klíčového materiálu (generování, obnova, distribuce, sledování využití, archivace, likvidace), sdílení klíčů a dlouhodobou dostupnost kryptografických objektů všech typů (veřejných a soukromých klíčů, certifikátů, symetrických klíčů a jiných forem „sdílené tajemství“). Zahrnuje tyto prvky:

a) řízené objekty;
b) atributy objektů;
c) protokolové operace.

a) Objekty

Z kategorie řízených objektů má šest kryptografický charakter (např. Symmetric Key). Šablona Template indikuje nekryptografický objekt, Policy Template specifikuje způsob použití šablony. Další objekty lze vytvářet jako typ Opaque (objekt, jehož vlastnosti nejsou v protokolu viditelné).

Přiřazovanými parametry jsou pro klíče jejich bloky a pro certifikáty jejich hodnoty. Bloky klíčů jsou struktury určené k zapouzdření informací asociovaných s kryptografickými klíči. Privátní klíče jsou zachyceny jako objekt podle standardu PKCS #1 či PKCS # 8 v kódování DER (Distinguished Encoding Rules) notace ASN.1[4] (Abstract Syntax Notation One, viz standard ITU-T X.208).

b) Atributy

Atributy obsahují metadata řízených objektů, přiřazovány jsou protokolovou operací Locate. Mohou být určeny při vytváření objektů, protokolovými operacemi (např. atribut Certificate Type může být nastaven operacemi Register, Certify nebo Re-certify) nebo klientem. Některé atributy lze přidávat, modifikovat i rušit. Mohou mít více hodnot (instancí), řadu hodnot může mít např. atribut Object Group objektu Symmetric Key.

Atributy lze rozdělit do tří skupin: popisující, „co“ je objekt, určující, jak objekt použít, a popisující ostatní vlastnosti.

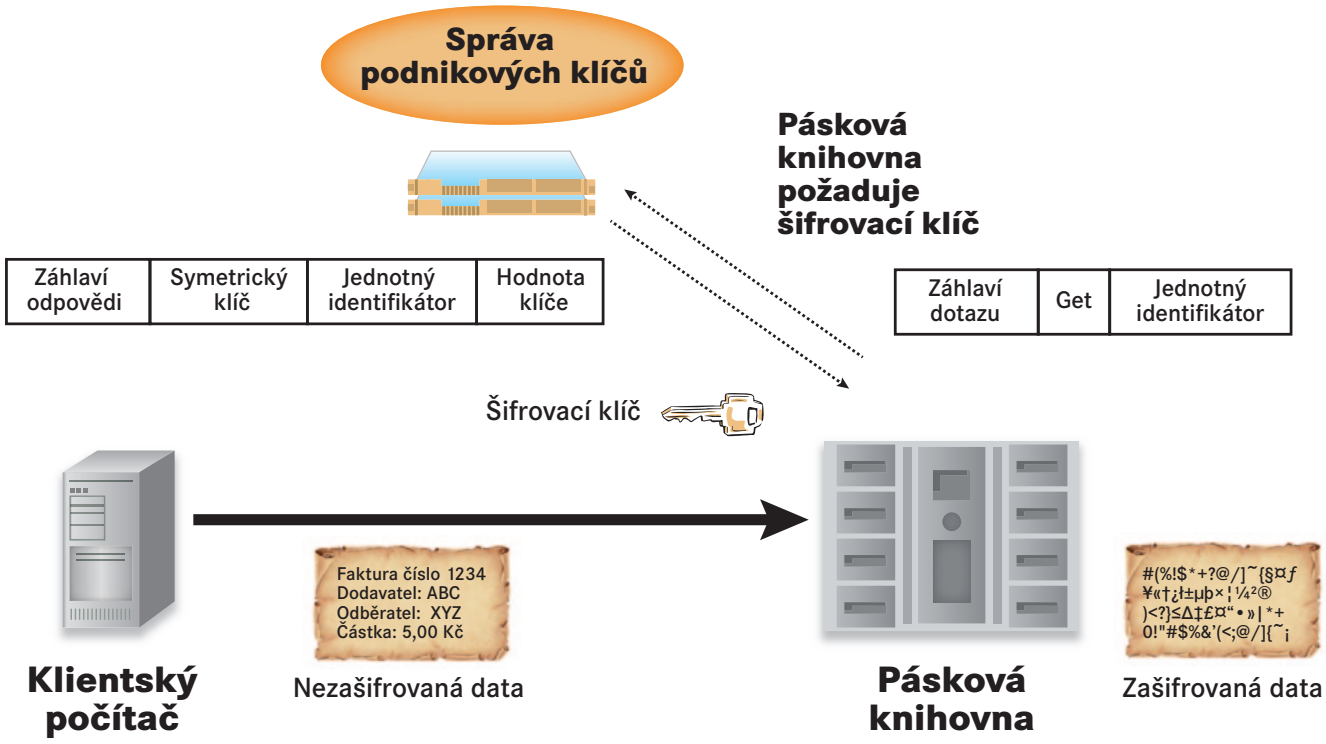
Pro přenos klíčů se používají objekty typu blok klíčů. Jako hodnota může být klíčový materiál s volitelnými atributy – Cryptographic Algorithm (např. AES, 3DES, RSA), Cryptographic Length (např. 128, 256, 2 048), Cryptographic Usage Mask (určuje použití klíče, např. Encrypt, Wrap Key či Export), Cryptographic Parameters (blíže identifikují použití klíče, např. mód šifrování – CBC, GCM, resp. NISTKey Wrap), State (např. Active), Dates (např. Activation Date) a Custom Attribute (např. se lze bránit proti přehrávání zpráv jejich označováním čísly – tzv. nounces).

Kódování zpráv je typu TTLV (Tag, Type, Length, Value) neboli „návěští, typ, délka, hodnota“, kde hodnotou může být rekurzivně další struktura typu TTLV. Na obr. 1 je ukázka kódování zprávy **`Get(symmetric key)`** předávající klíč o hodnotě **`0a33e83e-5b7a-4865-964a-8d1c3bbf9ae3`** identifikovaný identifikátorem **`0x420094`**.

c) Operace

Operace jsou typu klient – server. Existuje šest typů a slouží pro generování objektů, vyhledání a získání objektů, nastavení a čtení atributů, použití objektů, podporu volitelných operací a podporu asynchronních odpovědí.

[1] ASN.1 zná řadu typů kódování, DER je podmnožinou kódování BER (Basic Encoding Rules) a používá se např. v rámci protokolu X.509.



Obr. 2: Příklad výměny zpráv protokolu KMIP – knihovna pásek se zprávou Get obrací na podnikovou správu klíčů s požadavkem na symetrický klíč.

Zprávy protokolu

Základní operace jsou párové: dotaz klienta a odpověď serveru. Více párů typu dotaz – odpověď lze balit do dávek. Dotazy mohou být kladeny s využitím šablony atributů, struktura odpovědi však může být jiná, než jakou požaduje klient.

Používají se i nevyžádané zprávy ze strany serveru:

- zprávy, s jejichž pomocí server informuje o změnách hodnot atributů;
- zprávy, pomocí kterých server poskytuje klientovi nové objekty či atributy (indikuje se, zda nový objekt nahrazuje či ne ten existující).

I tyto zprávy lze sdružovat do dávek.

Zprávy jsou tvořeny záhlavím, zátěží (aktuální operace typu dotaz či odpověď) a případným rozšířením konkrétního výrobce systému KMIP. Pokud jsou zprávy posílány v dávkách, musí být v záhlaví uveden jejich počet. Implicitně je zpracování dávky uspořádané v pořadí položek, lze ale povolit i zpracování volné.

V záhlaví může být umístěna celá řada dalších volitelných parametrů, např. autentizační parametry či časové razítko. Některé parametry jsou jen specifické pro dotazy (např. omezení velikosti odpovědi či indikátor asynchronnosti), jiné pro odpověď (např. výsledek dotazu, a pokud je výsledkem chyba, lze požadovat i uvedení její příčiny). Ukázku nejtypičtější výměny zpráv v rámci komunikace klient – server, tj. vyžádání si symetrického klíče, poskytuje obr. 2.

Použité kryptografické funkce

Podle návrhu standardu [9] lze předávání zpráv zabezpečit povinně pomocí TLSv1.0, volitelně pomocí v1.1 (RFC 4346), resp.TLSv1.2 (RFC 5246), Není podporována žádná varianta SSL.

U varianty TLSv1.0 je povinnou kryptografickou sadou:

`TLS_RSA_WITH_AES_128_CBC_SHA`

V normě NIST 800-57 Part 3 [12] je v tabulce 4-1 až 4-4 uvedeno dalších několik desítek volitelných variant šifrovacích sad, z nichž nelze pro KMIP použít pouze variantu s nulovým šifrováním.

U varianty TLSv1.2 jsou povinné dvě kryptosady:

`TLS_RSA_WITH_AES_256_CBC_SHA256`

`TLS_RSA_WITH_AES_128_CBC_SHA256`

Volitelné sady jsou opět čerpány z normy NIST 800-57 Part 3. Na obr. 3 je uveden příklad specifikace kryptografického algoritmu a délky klíče.

Někdo může být zklamán tím, že si autoři ulehčují život recyklací již hotových kryptografických mechanismů. Ale tak se dnes moderní protokoly tvoří, nezačíná se od Adama. Předností autorů protokolu KMIP je, že nejde o jejich první

```
Tag: Attribute (0x420008), Type: Structure (0x01), Data:
   Tag: Attribute Name (0x42000A), Type: Text String (0x07), Data: Cryptographic Algorithm
   Tag: Attribute Value (0x42000B), Type: Enumeration (0x05), Data: 0x00000003 (AES)
Tag: Attribute (0x420008), Type: Structure (0x01), Data:
   Tag: Attribute Name (0x42000A), Type: Text String (0x07), Data: Cryptographic Length
   Tag: Attribute Value (0x42000B), Type: Integer (0x02), Data: 0x00000080 (128)
```

Obr. 3: Specifikace kryptografického algoritmu AES pro symetrické šifrování klíčem o délce 128 bitů.

práci na standardech a že se v kryptografii dobře orientují.

Závěr

ICT specialisté bojují proti únikům dat vyššími nároky na audit, ale ve stávajícím nepřehledném kryptografickém prostředí velkých firem, institucí a organizací je problémem ho zodpovědně provádět. Očekává se, že jednotný protokol správy klíčů provádění auditů ulehčí. Další pozitivní přínos lze předpokládat v oblasti snížení celkových nákladů na ICT.

Uvedli jsme si, co KMIP je, a nyní je vhodné uvést, co není. Především není API (Application Programming Interface), protože nedefinuje sadu služeb volaných aplikací. Není ani objektovým modelem s odpovídajícími metodami volanými aplikacemi. Nezávislost na velikosti paměti, výkonnosti procesoru atd. umožňuje, aby protokol KMIP používal libovolný kryptografický klient, od PDA po ta největší disková pole.

Na bázi protokolu KMIP dochází k „harmonizaci" protokolů pro správu klíčů. Např. pracovní skupina IEEE P1619 SISWG plánuje převzetí datových primitiv [10] a pracuje na mapování primitiv OASIS KMIP datových typů do odpovídajících typů XSD a datových typů programovacího jazyka C++ [11].

Je zřejmé, že se práce na KMIP překrývají s úsilím pracovních skupin IEEE P1619, EKMI i jiných, přestože se snaží s nimi být komplementární. Tak jako KMIP bude ovlivňovat specifikace jiných protokolů, jeho další vývoj nemůže být bez zpětné vazby. Např. by KMIP mohl mít variantu opírající se o uživatelsky přívětivý jazyk XML, který je použit ve specifikacích KEYPROV a EKMI. Také lze předpokládat, že v modulech HSM (Hardware Security Module) bude KMIP v budoucnu integrován se standardem IEEE 1619.3. Ještě je třeba podotknout, že jak se vzájemně nové protokoly překrývají, žádný nepokrývá finanční služby, což je škoda.

Jak bude KMIP rozvíjen dále? Autoři předpokládají, že poslední pracovní návrh protokolu bude v oficiální verzi 1 akceptován prakticky beze změn. Proto v době psaní tohoto článku již pracovali na verzi 1.1, která má být zveřejněna do konce tohoto roku. Příští rok má vyjít verze 2.

Jaroslav Dočkal
jaroslav.dockal@dsm.tate.cz

[5] IEEE P1619.3 Security in Storage Working Group (SISWG) řeší kryptografickou ochranu klíčů pro paměťová média.
[6] XML Schema Definition – schéma popisující strukturu XML dokumentu.
[7] IETF Provisioning of Symmetric Keys (KEYPROV) Working Group – její protokoly jsou omezeny na distribuci symetrických klíčů pro autentizační tokeny.
[8] Informace je od Roberta Haase, který to uvedl v mailu autorovi tohoto článku.

Doc. Ing. Jaroslav Dočkal, CSc.

Absolvent VDU Martin a Univerzity obrany, v současnosti docent téže univerzity, lektor Cisco akademie, externí školitel společnosti Hewlett-Packard a šéfredaktor DSM.

POUŽITÉ ZDROJE

[1] BALL, M. *Sun Microsystems, Proposed changes against P1619.3/D6 to use OASIS KMIP as the basis for the P1619.3 binary encoding*. Inc. IEEE P1619.3 Task group of the Security In Storage Working Group. Date: August 28, 2010. http://www.symantec.com/connect/articles/detecting-worms-and-abnormal-activities-netflow-part-1.
[2] BALL, M. *Sun Microsystems. Mapping OASIS KMIP to an XML WSDL for IEEE P1619.3*. Version 2, February 15, 2010. http://xml.coverpages.org/Ball-Mapping-OASIS-KMIP-to-XML-v2.pdf.
[3] BARKER E. etc. *NIST Special Publication 800-57 RECOMMENDATION FOR KEY MANAGEMENT*. Part 3: Application-Specific Key Management Guidance. U. S. Department of Commerce. December 2009.
[4] *Cover Pages Topic Document „Cryptographic Key Management"*. http://xml.coverpages.org/keyManagement.html.
[5] *Key Management Interoperability Protocol Specification Version 1.0*. Committee Draft 10 / Public Review 02, 18 March 2010. http://docs.oasis-open.org/kmip/spec/v1.0/cd10/kmip-spec-1.0-cd-10.html.
[6] *Key Management Interoperability Protocol Usage Guide Version 1.0*. Committee Draft 09 / Public Review 02. http://xml.coverpages.org/ni2009-02-27-a.html.
[7] *Key Management Interoperability Protocol Use Cases Version 1.0*. Committee Draft 09 / Public Review 02, 18 March 2010. http://docs.oasis-open.org/kmip/usecases/v1.0/cd09/kmip-usecases-1.0-cd-09.html.
[8] *Key Management Interoperability Protocol Profiles Version 1.0*. Committee Draft 05 / Public Review 02, 18 March 2010. http://docs.oasis-open.org/kmip/profiles/v1.0/cd05/kmip-profiles-1.0-cd-05.html.
[9] TOMHAVE, B. *Dysfunction Junction – Do Standards Function?* The ISSA Journal February 2010, Vol. 2, Issue 2, s. 12–16, 34.