Šifrování datových úložišť

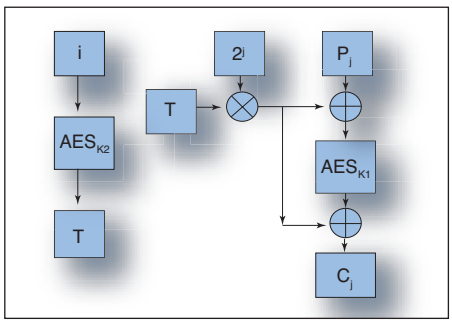
Ochraně dat při komunikaci je obvykle věnována velká pozornost. Stejná potřeba ovšem vzniká i u archivovaných dat. Zejména pokud jde o zálohy citlivých informačních systémů. Na velikosti firmy přitom mnoho nezáleží, vždy potřebujeme v reálném čase ukládat na záložní média ohromná množství dat. Ještě dnes se zálohy poměrně často ukládají nešifrované. Když se ztratí, jako tomu bylo například u Bank of America, Ameritrade nebo University of Berkeley, je z toho velký skandál a hlavně problém pro klienty, jejichž účty nebo osobní údaje jsou k dispozici organizovanému zločinu nebo vystaveny na Internetu.

RSJRPNFONF AF TUBOEBSEZ OFSFTJ VRMOPV PDISBOV EBU OB EJTLV
B QBTLBDI WAEZ KF EVMFAJUZ NPEFM ISPAFC

Obr. 1 Klasická substituce

Teoretický i praktický problém

Zašifrované ukládání dat se přitom až donedávna systematicky vůbec neřešilo, takže zaostávala i teorie. Chyběly jak účinné kryptografické algoritmy, tak standardy, které by je definovaly. Než byl přijat standard k šifrování disků, trvalo to čtyři roky. Nyní se připravuje standard k šifrování pásek a pro zacházení se šifrovacími klíči k těmto médiím. Začněme se šifrováním dat na pevných discích. Chráněný sektor rozdělíme na bloky po 16 bajtech a ty každý zvlášť zašifrujeme pomocí AES (s klíčem *K*). Je to tedy tajná substituce, která pracuje na úrovni symbolů, kterými jsou 128bitové bloky. Nyní si vzpomeneme na substituci na úrovni symbolů, kterými jsou písmena. Takovou substitucí se šifrovalo tisíce let, dnes víme, že pokud je šifrový text dost dlouhý, vztahy mezi šifrovými symboly odhalují vztahy mezi otevřenými symboly, a tím celou otevřenou zprávu. To můžeme vidět na příkladu zašifrované zprávy, která používá písmena *A* až *Z* na *obr. 1*. Brzy například zjistíme, že písmena *JPFBZ* jsou šifrové substituty samohlásek, neboť jejich výskyt odpovídá 40% výskytu samohlásek v otevřeném textu a pravidelnému střídání samohlásek a souhlásek. Jakmile 26 symbolů *A* až *Z* nahradíme 2^{128} symboly 0000...000 až 1111...111 (128bitové binární bloky) a aplikujeme na ně tajnou substituci (AES s klíčem *K*), budeme asi těžko takový zašifrovaný disk luštit jako původní substituci, protože vztahů mezi šifrovými symboly je velmi mnoho. Avšak přesto takový disk často vyzařuje informaci, která by neměla uniknout. A za druhé i zašifrovaný disk umožňuje manipulaci s daty tak, aby to stačilo útočníkovi. Například pokud na disku vidíme velmi mnoho bloků se stejnou (byť nesmyslnou) hodnotou, je nanejvýše pravděpodobné, že odpovídají prázdnému prostoru na disku. Útočník tak může odhadovat délku dat, může zjistit, zda od posledního náhledu byl na disk uložen nějaký nový soubor. Dále může libovolně vyměňovat 128bitové bloky mezi sebou, a pokud ví, kde jsou uložena důležitá data, může tak docílit jejich zajímavých záměn (třeba u databáze platů). U interaktivních útoků má možností ještě více. Typicky může systému zaslat zprávu (databázový záznam), zjistit její šifrový obraz a potom kdykoliv tento záznam do systému opět vložit (třeba zaslání obnosu na konto), přemazat s ním jiný „nevhodný“ záznam a podobně. Možná s údivem zjišťujeme, jaké nástroje útočník má. U každého systému je dobré si stanovit model hrozeb a říci, jaké situace a scénáře u konkrétního systému musíme uvažovat. Například nový standard pro šifrování disku (algoritmus XTS, viz dále) znemožňuje výměnu bloků, ale nedetekuje změnu bloku (klidně rozšifruje i nesmysly). Také nezabrání tomu, aby útočník nezaměnil nový obsah několika sektorů disku za (smysluplný) starý obsah. To lze využít například k návratu systému do místa před změnou (například výběr peněz), kterou chce útočník zakrýt. Nový standard používá algoritmus, který šifruje každých 16 bajtů na disku jiným způsobem, jinou šifrou, přičemž je ještě poměrně rychlý a nepotřebuje příliš mnoho paměti a nezvyšuje velikost dat. Neošetřuje však všechny modely hrozeb, na což je potřeba vždy myslet. Zejména je nutno zvážit zajištění integrity šifrovaných dat, tj. jejich ochranu proti neoprávněné modifikaci.



Obr. 2 Šifrování *j* tého bloku (128 bitů) v *i* tém sektoru disku algoritmem XTS-AES

Standardy

Nejznámějšími standardy pro šifrování ukládaných dat jsou standardy skupiny P1619 organizace IEEE. Standard P1619.0 definuje šifrování disků použitím algoritmu XTS-AES. Standard P1619.1 (draft) navrhuje šifrování pásek použitím algoritmu AES v modech Counter modus s CBC-MAC (CCM), Galois/Counter Mode (GCM, viz minulé číslo ST), CBC-HMAC-SHA a XTS-HMAC-SHA. Standard (draft) P1619.2 se zabývá šifrováním disků s použitím blokové šifry se širokým blokem (512 bitů). Také se pracuje na standardu klíčového hospodářství pro paměťová média (draft P1619.3).

Algoritmus XTS-AES

Tento šifrovací algoritmus vznikl modifikací a sloučením několika návrhů. Může mít tři délky klíče *K*: 256, 320 a 384 bitů. Klíč rozdělí na část *K1*, který použije jako klíč algoritmu AES-128, 192 nebo 256, a část *K2*, která má 128 bitů. Jak je vidět na *obr. 2*, šifruje každý blok (*j*=0, 1, ...) každého sektoru (*i*=0, 1, ...) odlišným algoritmem. *K2* se použije pro výpočet modifikační hodnoty pro každý 128bitový blok $T(i,j)=AES_{K2}(i).2^j$. Násobení probíhá v Galoisově tělese $GF(2^{128})$, určeném polynomem $x^{128}+x^7+x^2+x+1$. Hardwarově se výpočet hodnot $T(i,j)$ snadno realizuje pomocí posuvného registru. Na počátku se registr naplní hodnotou $AES_{K2}(i)$ a v každém taktu se získá hodnota $T(i,j)$ pro další *j*. Jak je vidět, šifrování každého bloku disku P_j je jiné. Nezabraňuje to všem útokům, ale alespoň těm nejčastějším. Proto má dobrý poměr cena/výkon. Poznamenejme, že umí šifrovat i sektory, které nejsou zarovnány na 128 bitů, například sektory o 520 bajtech, a to technikou kradení šifrového textu (podrobněji viz ST 12/2003). Předposlední úplný a poslední neúplný blok otevřeného textu se v tomto případě šifrují zvláštním způsobem.

Závěr

Dlouhou dobu nebyla standardizována ochrana ukládaných (zálohovaných) dat. V současné době je pro tyto účely v pokročilých pracovních verzích k dispozici už několik standardů. Připomeňme, že kromě dobrého standardu je vždy důležitý konkrétní model hrozeb.

Vlastimil Klíma, Tomáš Rosa,
v.klima@volny.cz, trosa@ebanka.cz

LITERATURA

[1] IEEE P1619.0: Standard for Cryptographic Protection of Data on Block-Oriented Storage Devices
[2] IEEE P1619.1: Standard for Authenticated Encryption with Length Expansion for Storage Devices
[3] IEEE P1619.2: Standard for Wide-Block Encryption for Shared Storage Media
[4] IEEE P1619.3: Standard for Key Management Infrastructure for Cryptographic Protection of Stored Data