Kryptologie pro praxi – zesílení hašovací funkce třídy SHA-1

Oblast digitálních podpisů se dostává do mírně krizového stavu, neboť za světově nejrozšířenější a nejpoužívanější funkce MD5 a SHA-1 už nedá nikdo ruku do ohně. Za MD5 už vůbec ne a u SHA-1 NIST od garancí upouští. Pochopitelně, když se za poslední rok a půl snížila bezpečnost SHA-1 více než 100 000 krát. A tak americký standardizační úřad NIST vydal doporučení přestat ji používat ihned, kde to lze, a jako standard by měla fungovat nejdéle do roku 2010. Pak už si odpovědnost za případné škody ponesou sami uživatelé. Teď všechny čeká nová etapa výměny za bezpečnější funkce třídy SHA-2.

Je kryptografie tak neschopná, že nedokáže nabídnout bezpečnou hašovací funkci, když je to klíč pro všechny digitální podpisy, elektronickou výměnu dat, protokoly a bůhvíco ještě? Problém je v tom, že vývojáři a uživatelé nechtěli vidět fakt, že kryptografické techniky zastarávají. Prodlužují jejich život tak dlouho, dokud je realita ke změně nedonutí. Žádné předvídání, žádná rezerva. A to je dnešní stav. Proto právě NIST vyvinul rozsáhlé mezinárodní úsilí k navržení nového konceptu hašovacích funkcí.

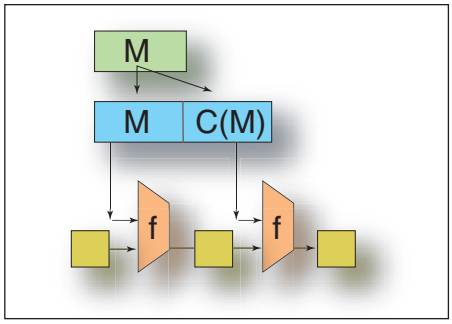
Celou situaci shrnuje motto „nedůvěřujte kryptologům“, které nabádá brát kryptografické techniky jako časově omezené bezpečnostní nástroje a starat se o jejich aktualizaci tak jako ve všem, co se týká bezpečnosti informačních systémů.

Vzorově to dokládá funkce MD5, která se za poslední rok a půl proměnila z „dost bezpečné hašovací funkce“ na ruinu. Nejnovější praktický výsledek tzv. tunelování hašovacích funkcí, je z března t.r. a umožňuje generovat její kolize v průměru za 17 sekund na obyčejném PC [3]. S tímto veřejně dostupným zdrojovým kódem lze pak dělat zajímavé triky, např. viz [4].

V dubnovém čísle jsme nabídli ještě záchranu těm, kdo nemají jinou možnost, než používat MD5 (například v uzavřených systémech). První úprava byla platná pro všechny hašovací funkce a spočívala v doplnění každého bloku zprávy o jeho kontrolní kód (*obr. 1*). Druhá úprava využívala hašovací kód několika instancí MD5 najednou, kde se jednotlivé MD5 liší inicializačními konstantami. To je také přijatelná záplata pro některé systémy. Dnes ohodnotíme bezpečnost takové konstrukce i pro SHA-1 a uvedeme ještě další možnosti.

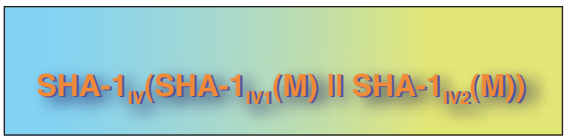
Bezpečnost k-násobné SHA-1

Uvažujme tedy hašovací funkci F, jejíž hašový kód $F(M)$ zprávy M se skládá ze zřetězení k hašových kódů $F_1(M) \,||\, F_2(M) \,||\, \ldots \,||\, F_k(M)$. Za F_i uvažujeme SHA-1_{IVi}, tj. funkci SHA-1 s inicializační konstantou IVi. Ještě před rokem a půl se všeobecně



Obr. 1 Obecné doporučení pro všechny hašovací funkce (ST 4/06)

myslelo, že složitost nalezení kolize takové funkce bude zhruba $2^{80} * 2^{80} * \ldots * 2^{80}=2^{80k}$, kde 2^{80} je, jak víme, složitost nalezení kolize SHA-1 narozeninovým paradoxem. Metody Wangové [1] však z čísla 80 nyní udělaly 63 a metody Jouxové [2] pak celý vzorec zdegradovaly na: 2^{63} (pro k=1), 2^{80} (pro k=2) a pouhých $2^{80+6,5(k-2)}$ pro k větší než 2. Abychom vrátili SHA-1 její původní zamýšlenou kvalitu, tj. složitost 2^{80}, museli bychom použít dvě zřetězení. Podobný vzorec platí i pro MD5, kde složitost je: 17 sekund (pro k=1), 2^{64} (pro k=2), pouhých $2^{64+6(k-2)}$ pro k větší než 2. Zároveň tím opravujeme numerickou chybičku, která se vloudila do článku v dubnovém čísle, kde uvedené odhady byly optimističtější.



Zachrání tento vzorec SHA-1 ?

Univerzální posilovač?

Existuje možnost jak vrátit hašovacím funkcím MD5 i SHA-1 původní kvalitu? Byla by to záchrana „nejhorších“ situací, kdy máme k dispozici pouze původní hašovací funkci a pro hašovací kód pouze původní délku. Zdá se, že taková úloha nebude mít řešení, ale má. Z dosud uvedeného vyplývá, že kdybychom mohli místo SHA-1(M) použít SHA-$1_{IV1}(M) \,||\, \text{SHA-}1_{IV2}(M)$, byla by složitost 2^{80} zpátky. Příliš dlouhý hašový kód můžeme zkrátit pomocí SHA-1! V tomto případě „zpráva“, kterou hašujeme originální hašovací funkcí, se skládá z vysoce závislé a strukturované zprávy SHA-$1_{IV1}(M) \,||\, \text{SHA-}1_{IV2}(M)$, kterou nelze libovolně „posunovat“ pro potřeby útoku. Proto (zatím) jediná cesta jak docílit kolize této konstrukce: SHA-1_{IV}(SHA-$1_{IV1}(M) \,||\, \text{SHA-}1_{IV2}(M)$) je narozeninovým paradoxem, a to dává složitost 2^{80}. Vidíme, že řešení se našlo. Podobně u MD5 by to byla místo původní MD5(M) haš $\text{MD5}_{IV}(\text{MD5}_{IV1}(M) \,||\, \text{MD5}_{IV2}(M))$, která jí vrací složitost 2^{64}. Jednu nevýhodu tento přístup má. Je postaven na funkcích, u nichž byly nalezeny slabiny a pravděpodobně ještě nějaké nalezeny budou.

Další alternativa SHA-1

Další alternativou k posílení SHA-1 je upravit hašovací funkci uvnitř. Proto vznikl pracovní návrh nového standardu SHA1-IME. Podrobný popis SHA1-IME naleznete v [5], programová realizace této funkce se od SHA-1 liší jen v jednom řádku navíc, v tzv. lepší expanzi zprávy.

Závěr

Pokud bychom před rokem a půl řekli, že dnes bude možné během několika sekund vytvářet kolize hašovací funkce MD5 na notebooku, asi by nám nikdo nevěřil. Než obdrží takový výsledek pro SHA-1, může to trvat měsíc, nebo deset let, to nikdo neví. Do žádných hurá výměn nikoho nenutíme, uvedli jsme jen trochu více informací, abyste si mohli ohodnotit bezpečnost a upřesnit riziko dalšího používání té či oné hašovací funkce nebo zvolit nějaké jiné alternativy.

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

LITERATURA

[1] Wang, X., Feng, D., Lai, X., Yu, H.: Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD, rump session, CRYPTO 2004,

[2] Joux, A.: Multicollisions in iterated hash functions. Application to cascaded constructions. Crypto 2004, pp. 306–316.

[3] Klima, V.: Tunnels in Hash Functions: MD5 Collisions Within a Minute, v češtině na http://cryptography.hyperlink.cz/2006/tunely.pdf, zdrojové kódy na http://cryptography.hyperlink.cz/2006/web_version_1.zip

[4] Domácí stránka projektu kolizí: http://cryptography.hyperlink.cz/2004/kolize_hash.htm

[5] SHA1-IME, Internet draft, November 2005, <draft-irtf-cfrg-sha1-ime-00.txt>,

[6] E-archivy http://cryptography.hyperlink.cz, http://crypto.hyperlink.cz/