Portál pre odborné publikovanie ISSN 1338-0087

Zachytenie šifrovacích kľúčov pomocou analýzy operačnej pamäte a čitateľné heslá v pamäti I.

Pondelík Matej · Elektrotechnika, Študentské práce

27.10.2010

Tento článok sa zaoberá šifrovaním pevných diskov a zachytením šifrovacích hesiel použitých šifrou AES z operačnej pamäte, ktorej výpis môžeme získať rôznymi technikami. Práca sa čiastočne zaoberá aj heslami, ktoré sú uložené v operačnej pamäti v čitateľnej forme. Cieľ práce je zameraný hlavne na linuxovú distribúciu Ubuntu a šifrovací nástroj TrueCrypt, no bude sa zaoberať aj operačným systémom Windows.

1. Úvod

Všeobecne platí, že šifrovanie dát je potrebné hlavne pre tých, ktorí majú čo skrývať. Používajú ho firmy, bežní užívatelia, ale aj teroristické organizácie (napríklad Al-Káida používa softvér Mujahideen Secrets 2). Šifrovanie je dvojsečná zbraň. Na jednej strane chráni naše citlivé údaje, ktoré majú často najvyššiu hodnotu, no na strane druhej umožňuje kriminálnikom ukrývať dáta, ktoré by ich mohli usvedčiť. V súčasnosti je jedným z najlepších voľne dostupných nástrojov na šifrovanie je TrueCrypt. Je to open-source softvér na šifrovanie dát v reálnom čase, ktorý je kompatibilný s platformami Linux, Windows aj Mac OS X. Našim cieľom je zachytiť šifrovacie heslá, ktoré tento verejne rozšírený softvér používa pri šifrovaní. Bližšie sa mu budeme venovať v časti 2.

Ďalej sa budeme zaoberať metódami, ktorými sa dá získať výpis obsahu celej operačnej pamäte, alebo len pamäte pridelenej konkrétnemu procesu. Následne budeme tento výpis analyzovať a získavať z neho dôležité údaje a šifrovacie kľúče. Koncom marca tohto roku vyšla nová verzia softvéru Passware Password Recovery Kit Forensic v.9.7 od firmy Passware, Inc.. Je to prvý komerčný softvér, ktorý dokáže dešifrovať disky, ktoré sú vytvorené TrueCrypt-om alebo BitLocker-om. Na to však potrebuje obraz operačnej pamäte počítača, na ktorom je pripojený šifrovaný disk. Bližšie sa týmto softvérom budeme zaoberať v časti 7. Nakoniec si povieme o heslách, ktoré sú uložené v operačnej pamäti v čitateľnej forme pod distribúciou Ubuntu.

2. TrueCrypt

TrueCrypt je softvér na šifrovanie dát v reálnom čase (OTFE – On-the-fly encryption), čo znamená, že dáta sú šifrované alebo dešifrované v momente, keď sa ukladajú alebo nahrávajú. Šifrovanie a dešifrovanie je vykonávané automaticky bez zásahu užívateľa a žiadne dáta uložené na zašifrovanom zväzku (médium na ukladanie dát, napr.: USB, pevný disk, vytvorená partícia, celý súborový systém, atď.) nie je možné čítať

(dešifrovať) bez použitia správneho hesla. Na disku sa nachádzajú len zašifrované dáta a dešifrovanie/šifrovanie prebieha priamo v operačnej pamäti.

2.1 Možnosti TrueCrypt-u

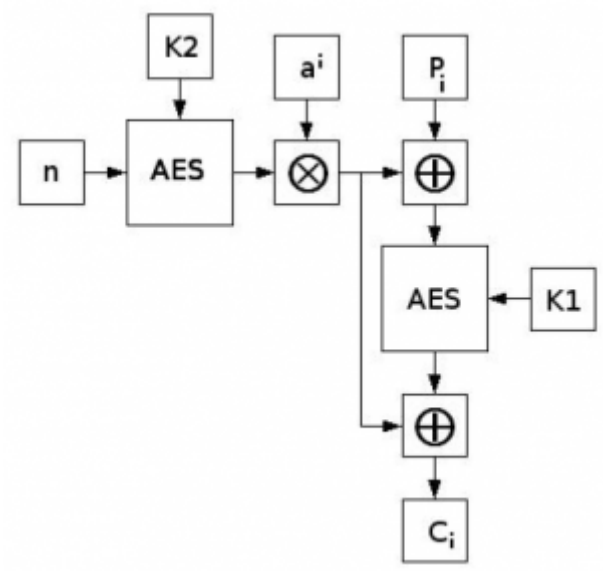
TrueCrypt [1] nám umožňuje vytvoriť šifrovaný súbor – kontajner, ktorý sa potom pripojí k systému ako virtuálny disk. Môžeme s ním šifrovať celé partície, prípadne celý súborový systém. TrueCrypt môže vytvoriť takzvaný skrytý zväzok ("hidden volume"), ktorý umožňuje pridať do jedného šifrovaného zväzku ešte jeden, ktorý môžme považovať za tajný. Takto vytvorený zväzok bude mať dve prístupové heslá. Jedno pre prístup k dátam, v prvej štandardnej šifrovanej oblasti a druhé k prístupu ku skrytej oblasti.

V prípade zadania prvého hesla sa bude javiť oblasť skrytého zväzku ako voľné miesto vyplnené náhodnými dátami (voľné miesto na každom TrueCrypt disku je vyplnené náhodnými dátami). V prípade, že je užívateľ nútený vydať heslo, či už pod hrozbou násilia alebo zo zákona, môže odšifrovať len prvú oblasť a skrytá oblasť je podľa autorov TrueCrypt-u neidentifikovateľná. Ako analogický príklad uvádzame Veľkú Britániu, kde je každý, kto používa šifrovanie k zabezpečovaniu svojich dát, povinný údaje na požiadanie britskej polície dešifrovať.

2.2 Šifrovacie algoritmy používané TrueCrypt-om

TrueCrypt používa na šifrovanie algoritmy AES, Serpent, Twofish a kaskádové šifrovanie kombináciou týchto algoritmov (AES-Twofish, AES-Twofish-Serpent, Serpent-AES, Serpent-Twofish-AES, Twofish-Serpent). Veľkosť kľúča je pre každý algoritmus 256 bitov, veľkosť šifrovaného bloku je 128 bitov a šifrovací mód je XTS (od verzie 5.0).

2.2.1. Šifrovací mód XTS pre AES



Obr. 1. Bloková shéma XTS-AES.

$C_i = EK1(P_i \oplus (EK2(n) \otimes a_i)) \oplus (EK2(n) \otimes a_i)$

$\otimes$ – násobenie dvoch polynómov nad binárnym poľom GF (2) modulo $x128+x7+x2+x+1$,

K1 a K2 – primárny a sekundárny kľúč (256 bitov),
i – index bloku šifry vzhľadom na blok dát, začínajúci od i=0,
n – index bloku dát vzhľadom na K1, začínajúci od n=0,
a – primitívny člen GF (2128) .
Veľkosť každého bloku dát je 512 bajtov, bez ohľadu na veľkosť sektora. Viac informácií v [2].

3. Metódy získania výpisu operačnej pamäte

Ako sme spomínali v predchádzajúcej časti, pri šifrovaní v reálnom čase prebieha šifrovanie len v operačnej pamäti. To znamená, že sa tam musia nachádzať aj šifrovacie heslá, ktoré chceme získať. Na získanie výpisu obsahu operačnej pamäte existuje niekoľko metód. Líšia sa vzhladom na operačný systém, či vybavenie potrebné k získaniu výpisu. Metódy, ktoré sú pre náš zámer najpodstatnejšie popisujeme v podkapitolách.

3.1. /proc/[pid]/mem

Nasledujúca technika je použiteľná na počítačoch s operačným systémom Linux. Pomocou tohoto postupu sme schopní získať presný obsah operačnej pamäte, ku ktorej pristupuje daný proces.
/proc je pseudo-súborový systém, ktorý slúži ako interfejs dátovej štruktúry kernelu. Okrem iného sa tu nachádzajú číselné subadresáre, pre každý bežiaci proces: /proc/[pid]/. Pre nás sú prioritné dva súbory:
proc/[pid]/maps – tento súbor obsahuje aktuálne namapované pamäťové sektory a ich prístupové práva, nachádzajú sa tu pre nás potrebné adresné miesta a názvy súborov, využívajúce tieto adresné miesta;
proc/[pid]/mem – tento súbor môže byť použitý k prístupu ku stránkam pamäte procesu pomocou funkcii open, read, lseek alebo fseek.

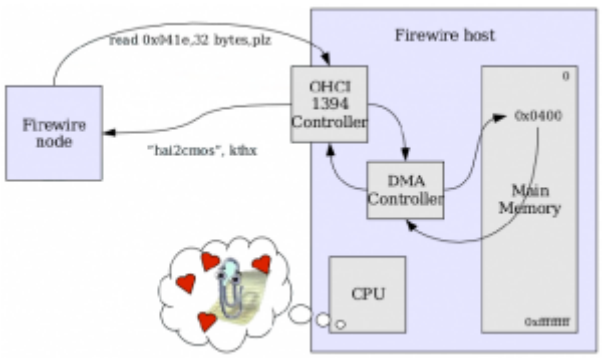
Pre prístup k týmto súborom však potrebujeme rootovké práva. Predpokladajme, že máme spustený program TrueCrypt a pomocou neho pripojený šifrovaný oddiel. Aby sme získali výpis pamäte procesov, ktoré TrueCrypt spustil budeme postupovať nasledovne:

1. Ako prvé musíme zistiť PID porcesu, ktorého výpis pamäte chceme analyzovať, v našom prípade je potrebné získať PID procesu TrueCrypt. Ten môžeme nájsť nasledovne:
   ```
   root@pc:/# ps -e | grep true
   5737 ? 00:00:36 truecrypt
   5738 ? 00:00:00 truecrypt
   5781 ? 00:00:00 truecrypt
   5783 ? 00:00:00 truecrypt
   5785 ? 00:00:35 truecrypt
   ```
   Ako môžeme vidieť, TrueCrypt vytvoril päť procesov. Prvé tri bežia okamžite po jeho spustení, posledné dva sa vytvoria až keď pomocou neho pripojíme šifrované médium. Pre získanie šifrovacích hesiel bude pre nás podstatný PID posleného procesu.
2. Pomocou zisteného PID teraz môžeme získať adresy stránok v pamäti, ku ktorým pristupuje proces. Tie získame zo súboru /proc/[pid]/maps, v našom prípade to bude /proc/5785/maps.

3. Teraz môžeme prejsť priamo ku kopírovaniu pamäti, ku ktorej pristupuje proces. Budeme pracovať so súborom /proc/5785/mem. Obmedzené práva nám však umožňujú užívať súbor iba na čítanie. Predpokladajme, že chceme získať výpis pamäťového priestoru haldy (heap) tohto procesu, v ktorej ako sme zistili sa nachádzajú aj šifrovacie heslá, k tomuto sa však vrátime neskôr. Zo súboru maps sme zistili že halda sa nachádza v operačnej pamäti na rozsahu adries 01749000-019c7000.
   - najprv musíme pozastaviť bežiaci proces, ktorého výpis pamäte chceme získať
     ```
     ptrace(PTRACE_ATTACH,pid,0,0);
     ```
   - otvoríme súbor "/proc//mem" na čítanie
     ```
     mem = fopen ("/proc/4414/mem", "r");
     ```
   - v súbore sa nastavíme na adresu, kde sa začína halda – teda 01749000
     ```
     fseek (mem, mem_start, SEEK_SET);
     ```
   - teraz môžeme prejsť ku kopírovaniu obsahu haldy (01749000-019c7000)
     ```
     while (mem_start != mem_end)
     {
     fread(&c, 1, 1, mem);
     fprintf (dump, "%c", (unsigned char)c);
     mem_start++;
     }
     ```
   - znovu spustíme stopnutý proces
     ```
     ptrace (PTRACE_DETACH, pid, 0, 0);
     ```
4. Výsledný súbor, ktorý obsahuje výpis haldy procesu si môžeme prezrieť v ľubovoľnom hexadecimálnom editore.

3.2 FireWire

Rozhranie FireWire bolo vytvorené spoločnosťou Apple. V roku 1995 bolo normalizované ako IEEE 1394. Toto rozhranie sa používa najmä pri spracovaní zvuku a videa, kvôli jeho vysokej efektívnej rýchlosti. Zariadenia prepojené cez rozhranie FireWire komunikujú medzi sebou prostredníctvom priameho prístupu do pamäte. Ak sa k počítaču obeti cez toto rozhranie pripojí útočník, má možnosť skopírovať obsah operačnej pamäte. Útočník má plný prístup k operačnej pamäti, čiže z nej môže nie len čítať ale aj do nej zapisovať. Toto sa deje na hardvérovej úrovni bez vedomia a zásahu operačného systému.



Obr. 2. fungovanie FireWire[4], Adam Boileau.

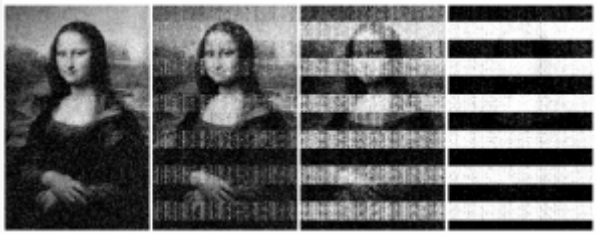
Túto bezpečnostnú slabinu, respektíve vlastnosť rozhrania FireWire prezentoval najskôr Maximillian Dornsief v roku 2004, v dokumente: "0wned by an iPod" [3], kde demonštroval ako môže zariadenie pomocou DMA čítať a zapisovať do pamäte za

pomoci iného počítača s Mac OS X, BSD alebo Linuxom. Neskôr v roku 2006 Adam Boileau zo Security-Assessment.com prezentoval “Hit By A Bus: Physical Access Attacks with Firewire”, kde rozšíril predchádzajúcu prácu a upriamil sa na Windows XP [4].

S nástrojmi, ktoré A. Boileau napísal v jazyku python, sme schopní pomocou počítača s Linuxom skopírovať obsah operačnej pamäte iného počítača, obísť prihlasovanie vo Windows, vložiť a spustiť proces v pamäti, ktorý sa nenachádza na disku cieľového počítača alebo zistiť posledných 16 bajtov z bufera klávesnice, ktoré boli prijaté BIOS-om.

3.3 Coldboot attack

Táto technika je založená na tom, že väčšina dnešných počítačov používa operačnú pamäť typu DRAM, t.j. dynamická pamäť. Táto pamäť uchováva náboj, a teda aj dáta v nej uložené po určitú dobu a potom sa začne postupne vybíjať, preto ju treba v pravidelných intervaloch obnovovať. Po vypnutí počítača sa očakáva, že pamäť RAM bude okamžite vymazaná. V praxi to však vo väčšine prípadov vyzerá tak, že obsah pamäte sa vymazáva postupne v priebehu niekoľkých desiatok sekúnd až minút ako to znázorňuje nasledujúci obrázok. Dáta uložené v pamäti sa po určitú dobu zachovajú dokonca aj po fyzickom odpojení pamätí z matičnej dosky. Pre dlhšie uchovanie dát môžeme pamäte zmraziť mraziacim sprejom.



Obr. 3. bitmapový obrázok uložený v RAM po odpojení napájania. Obrázky zľava po 5 sekundách (obrázok je nerozoznateľný od originálu), 30 sekundách, 60 sekundách a 5 minútach [5]

Ohrozené počítače sú tie, ktoré sú ponechané bez dozoru zapnuté, tesne po vypnutí, v stave hibernácie alebo v úspornom režime (v spáku), podobne ako je to pri možnosti získania obsahu pamäte cez rozhranie firewire. Tento fakt umožňuje prípadnému útočníkovi s fyzickým prístupom k počítaču svojej obeti, pripojiť externé pamäťové zariadenie na ktorom sa nachádza kód, ktorý je schopný po reštarte počítača a nabootovaní z tohto média skopírovať obsah pamäte RAM, prípadne môže útočník odpojiť pamäť a pripojiť ju na vopred pripravený počítač, ktorý skopíruje jej obsah.

Týmto spôsobom sa podrobne zaoberajú vývojári z Princetonskej Univerzity, Electronic Frontier Foundation a Wind River System [5]. V rámci tohto projektu vytvorili programy na získanie a rekonštrukciu (v prípade straty niekoľkých bitov) hesiel AES a RSA z takto získaného obsahu pamäte [6]. No ako sme zistili, ich program na vyhľadanie hesiel nie je vždy účinný. Viackrát sa stalo, že nenašiel ani jedno šifrovacie heslo.

3.4. Iné metódy

Pod operačným systémom Windows existuje viacero nástrojov na získanie výpisu operačnej pamäte, no hrozí však riziko pádu systému a naviac tieto výpisy nie sú vždy presné. Pri hibernácii Windows vytvára súbor hiberfil.sys, ktorý však tiež nie je presným obrazom operačnej pamäte. Jedna spoľahlivá metóda je vyvolať takzvanú "modrú obrazovku", pri ktorej sa vytvára presná kópia operačnej pamäte. Toto sa dá docieliť úpravou registrov[7].

Pokračovanie článku nájdete na našich stránkach v piatok 05.11.2010. V článku budú ukážky čítania šifrovacích klúčov v prostrediach Ubuntu a Windows.

Odkazy na literatúru

1. TrueCrypt, "TrueCrypt Home Page," 2010; http://www.truecrypt.org/docs/
2. Dworkin, M., Recommendation for Block Cipher Modes of Operation: The XTS-AES Mode for Confidentiality on Storage Devices, NIST Special Publication 800-3E, January 2010, Dostupné na internete: http://csrc.nist.gov/publications/nistpubs/800-38E/nist-sp-800-38E.pdf
3. Dornseif, M., "0wned by an iPod" Laboratory for Dependable Distributed Systems, PacSec 2004, Dostupné na internete: http://md.hudora.de/presentations/firewire/PacSec2004.pdf
4. Boileau, A., "Hit By A Bus: Physical Access Attacks with Firewire" Security-Assessment.com, Ruxcon 2006, Dostupné na internete: http://www.storm.net.nz/static/files/ab_firewire_rux2k6-final.pdf
5. Alex, J.H., Schoen, S.D., Heninger, N., Clarkson, W., Paul, W., Calandrino, J.A., Feldman, A.J., Appelbaum, J., Felten, E.W., Last we remember: cold-boot attacks on encryption keys. February 2008, Dostupné na internete: http://citp.princeton.edu/pub/coldboot.pdf
6. The Center for Information Technology Policy, Princeton University, http://citp.princeton.edu/memory/code/
7. wikihow, "wikihow Home Page," 2010; http://www.wikihow.com/Force-a-Blue-Screen-in-Windows
8. TrueCrypt, "TrueCrypt Documentation," 2010; http://www.truecrypt.org/docs/
9. Pasware, Inc., "Passware Kit Forensic 9.7," 2010; http://www.lostpassword.com/kit-forensic.htm

---

Spoluautorom článku je Ing. Štefan Balogh, Fakulta elektrotechniky a informatiky,Katedra aplikovanej informatiky a výpočtovej techniky, Slovenská technická univerzita