Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Varnost gesel

Seminarska naloga pri predmetu
Kriptografija in računalniška varnost

Tomaž Hiti

Mentor:
prof. dr. Aleksander Jurišić

Ljubljana, september 2006

POVZETEK

Z vnašanjem gesel se srečujemo vsakodnevno, zato je pomembno, da se naučimo pravilno ravnati z njimi. Šibka gesla lahko postavijo na kocko visoko varnost v marsikaterem informacijskem sitemu. Z osvojitvijo določenih tehnik za izdelavo močnih gesel in onemogočanjem zastarelih avtentikacijskih metod, lahko veliko pripomoremo k večji varnosti gesel.

Poglavje 1

Uvod

Vdori v sistem, nepooblaščena uporaba in zlorabe so pogosto posledica preveč enostavnih gesel, ki si jih uporabniki določajo sami. Z namenskimi programi za odkrivanje gesel lahko napadalci preverijo več kot 1.000.000 gesel v sekundi, pri generiranju pa so zelo iznajdljivi - uporabljajo celo slovarje besed in sezname najpogostejših gesel.

Geslo je niz znakov uporabljeno za namene avtentikacije. Uporabljamo ga za prijavo v operacijski sistem, za dostop do resursov, za prijavo v elektronsko pošto, itd. Skratka z gesli se srečujemo vsak dan. To je tudi razlog zakaj moramo posvečati več pozornosti varnosti gesel.

Šibka gesla postavljajo na kocko visoko varnost v marsikaterem informacijskem sitemu. Z osvojitvijo določenih tehnik za izdelavo močnih gesel in onemogočanjem določenih zastarelih avtentikacijskih metod, lahko veliko pripomoremo k večji varnosti gesel.

V drugem poglavju bomo podrobneje spoznali dinamična gesla in naprave za generiranje le-teh. Nadaljevali bomo s pregledom avtentikacijskih metod za Windows operacijske sisteme in njihovih zgoščevalnih funkcij (ang. hash function). Spoznali bomo dejstvo, da kar je pred 15 leti veljalo za dobro enkripcijo je danes mogoče razbiti v nekaj urah. V četrtem poglavju se bomo osredotočili na ranljivost gesel. Našteli bomo tipe napadov in za to namenska orodja. Nazadnje si bomo ogledali kako izdelati močna gesla in kaj lahko storimo za večjo varnost le-teh.

Poglavje 2

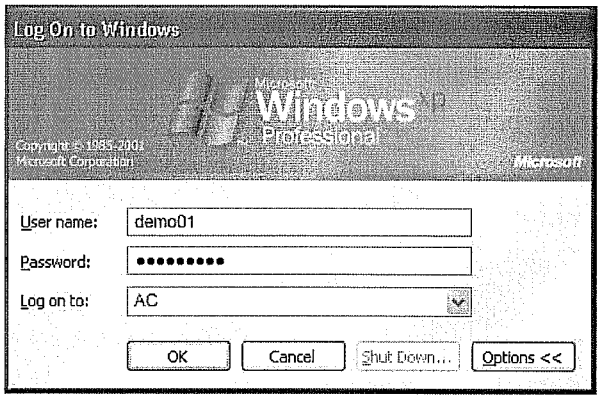
Gesla

2.1 Splošno

Gesla v grobem delimo na statična in dinamična. Eden od odločilnih korakov pri zaščiti omrežja podjetja je odprava statičnih gesel. Razlaga zakaj je temu tako, sledi v nadaljevanju.

2.2 Statična gesla

Statična (nespremenljiva) gesla so običajno shranjena na trdem disku. Uporabnik se najprej sreča z geslom ob prijavi v operacijski sistem. Ta od uporabnika zahteva, da se ob prijavi avtenticira z uporabniškim imenom in geslom.



Slika 1: Prijavno okno v WindowsXP OS

Težava je v tem, da si uporabniki radi izberejo preprosta gesla - taka, ki si jih bodo zlahka zapomnili. Tudi napadalci se tega zavedajo in to tudi s pridom izkoriščajo. Boljša alternativa statičnim geslom so dinamična gesla.

2.3 Dinamična gesla

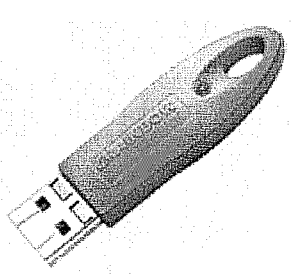
Druga vrsta gesel pa so dinamična (spremenljiva) gesla. Veljavna so samo za eno prijavo oz. avtentikacijo. Od tod tudi ime enkratna gesla. Ta gesla niso shranjena na lokalnem disku in kar je za uporabnika najpomembenjše - teh gesel si nam ti potrebno zapomniti. Napadalcu pa smo vsaj delno otežili krajo gesel, saj se gesla ne hranijo na lokalnem disku. Sledijo podrobnejše informacije o namenski strojni opremi.

2.4 Strojna oprema

Za generiranje dinamičnih gesel (enkratna gesla) so na tržišču majhne naprave, ti. žetoni (ang. tokens), ki predstavljajo v povezavi z ustreznim avtentikacijskim strežnikom zmogljivo rešitev za avtentikacijo. Glavna prednost uporabe žetona je preprosta uporaba, saj od uporabnika ne zahteva nobenega dodatnega nastavljanja in nobene programske opreme. Uporabnik vpiše le PIN in z enkratnim pritiskom na gumb pošlje zahtevo na avtentikacijski strežnik. Strežnik v primeru pravilne avtentikacije uporabnika, mu pošlje nazaj enkratno geslo.

Poznamo pa tudi druge vrste naprav, ki jih lahko uporabimo za avtentikacijo, dešifriranje in digitalno podpisovanje:

a) USB ključ



Majhna, priročna in prenosna naprava, ki združuje bralnik USB in pametno kartico. Za delovanje so potrebna USB vrata in ustrezen programski paket proizvajalca. Uporabnikom omogoča hranjenje identifikacijskih sredstev (ang. credentials), ki so potrebna za avtentikacijo, dešifriranje in digitalno podpisovanje sporočil (npr. elektronske pošte, datotek ipd.). Pred uporabo zahteva vnos PIN-a[1].

b) Pametna kartica

Pametna kartica je majhen računalnik z lastno centralno procesno enoto, kriptografsko procesno enoto, delovnim in neizbrisljivim pomnilnikom. Funkcionalnosti so iste kot pri USB ključu. Pred uporabo zahteva vnos PIN-a.

[1] PIN - Personal Identification Number

Poglavje 3

Avtentikacijske metode

3.1 Splošno

Kar je pred 15 leti veljajo za dobro enkripcijo je danes mogoče razbiti v nekaj urah. Z letom 1989 so pri Microsoftovih operacijskih sistemih začeli uporabljati LAN Manager avtentikacijsko metodo (v nadaljevanju LM). Ta metoda je za enkripcijo gesel uporabljala LM hash zgoščevalno funkcijo. Za takratno procesorsko moč je bilo razbijanje le-te izjemno težek problem. Danes pa temu ni več tako.

3.2 Avtentikacija s shranjenim geslom

Spodaj naštete avtentikacijske metode delujejo tako, da shranjujejo hash vrednosti gesel na lokalen disk.

a) LAN Manager (LM) - LM hash ima na žalost več minusov kot plusov. Njegova prva pomanjkljivost je omejevanje dolžine gesla - dovoljuje največ 14 znakov. Deluje tako, da pred enkripcijo geslo pretvori v velike tiskane črke, razpolovi geslo na dva dela po 7 znakov (2 DES[2] ključa), ter nato vsak del posebej kriptira z DES kriptosistemom [1]. Dobljeni 8B hash vrednosti na koncu združi v 16B hash vrednost. V veliki večini pri-

[2] Data Encryption Standard - simetrični kriptosistem

merov je dovolj, da napadalec napade oz. razbije le prvi del hash-a. Slednje velja še posebej, če je geslo krajše od 14 znakov. Novejše različice sistema Windows (2000/XP/2003) še vedno shranjujejo gesla z LM hash zgoščevalno funkcijo (zaradi kompatibilnosti za nazaj) in na zahtevo odjemalca še vedno opravlja LM avtentikacijo. Microsoft napoveduje, da bo s prihodom novega operacijskega sistema - Windows Vista, dokončno odpravljena LM avtentikacija. V podpoglavju 5.3 bomo podali napotke kako onemogočiti LM avtentikacijo.

b) NT LAN Manager v1 (prvo poimenovanje je bilo NTLM, ob prihodu izboljšane različice pa so jo preimenovali v NTLMv1) - avtentikacijska metoda NTLM je izboljšana različica LM. Izboljšana je varnost povezave med odjemalci in strežniki operacijskega sistema Windows. Odpravljeni sta 2 glavni pomanjkljivosti - geslo se ne pretvori več v velike tiskane črke in se kriptira enem kosu. Ampak tudi s to metodo naše geslo ni povsem varno. Na tržišču je moč kupiti strojni dekodirnik, ki razbije LM ali NTLM hash v 3 do 6 dnevih in to ne glede na dolžino gesla!
c) NT LAN Manager v2 (NTLMv2) - v okviru nenehnih prizadevanj, da bi izdelali varnejšo avtentikacijsko metodo, je Microsoft razvil izboljšano različico, imenovano NTLM različica 2. Zasnovana je na principu HMAC[3] z uporabo MD5 algoritma ali pa SHA-1 [1]. Avtentikacijska shema MAC je široko uporabljena metoda za ugotavljanje avtentičnosti sporočil, ki se pošiljajo po komunikacijskem kanalu med odjemalcem in strežnikom. Uporaba NTLMv2 avtentikacijske metode je za današnje sisteme priporočljiva, saj zelo izboljšuje varnost gesel in mehanizme varnostne seje.

Preden čistopis vstopi v zgoščevalno funkcijo se mu doda ti. *salt*. Salt je naključna vrednost generirana s pomočjo sistemske ure. Tako kriptirana hash vrednost se imenuje začinjena hash vrednost (ang. salted hash). Najprej so jo začeli uporabljati v UNIX operacijskih sistemih. Microsoft je salt začel uporabljati šele s prihodom NTLMv2. Prvi namen salt-a je bil preprečiti, da bi 2

[3] Keyed-Hashing for Message Authentication

uporabnika, ki bi izbrala isto geslo, imela isto hash vrednost. Na začetku je bil salt 12-bitni, tp. 4096 različnih hash vrednosti za eno geslo. Danes pa je salt precej daljši (32/64/128-bitni). S tem načinom kriptiranja gesel otežimo napad. Tabela se poveča za faktor 2^n (n je dolžina salt-a v bitih), saj ima vsaka beseda oz. geslo 2^n različnih hash vrednosti.

Poglavje 4

Ranljivost gesel

4.1 Splošno

Glede na ranljivost ločimo dve vrsti gesel: šibka gesla in močna gesla. Med šibka gesla štejemo vse tiste nize znakov, ki ustrezajo vsaj eni od naslednjih kriterijev: kratek niz, zaporednje tipk, abeceda, privzeta gesla[4], osebna imena, imena domačih živali, rojstni dnevi, povezava z uporabniškim imenom, itd. Med močna oz. robustna gesla pa štejemo vse tiste nize znakov, ki dosegajo določeno kompleksnost. Ta prag kompleksnosti je težko definirati. Obstajajo pa določene dobre prakse in priporočila za »izdelavo« močnih gesel. Več na to temo v podpoglavju 5.1.

Naš računalnik oz. širše gledano - informacijski sistem je izpostavljen vsakodnevnim nevarnostim. Naša dolžnost je, da poskrbimo za kar največjo stopnjo varnosti. Bodisi da nameščamo varnostne popravke, da dnevno osvežujemo antivirusno bazo, da ne odpiramo priponk čigar pošiljatelja ne poznamo, da imamo močno geslo itd. Organizacija CERT/CC[5] ocenjuje, da so za 80% vdorov v omrežja kriva šibka gesla oz. napačno ravnanje z gesli. Torej v nadaljevanju je opisana problematika o ranljivosti gesel.

---

[4] obsežen seznam privzetih gesel: http://www.phenoelit.de/dpl/dpl.html

[5] Computer Emergency Response Team / Coordination Center

4.2 Najbolj pogosti primeri gesel

Na žalost so najbolj pogosti primeri gesel, prav šibka gesla:

a) kratek niz (tj. niz dolg do štiri znakov)
b) zaporednje tipk na tipkovnici (qwert, asdf, 1234)
c) abeceda (abcd, dcba)
d) privzeta gesla (brez gesla, password, admin)
e) imena (osebna imena: anita, bojan; imena domačih živali: pluton)
f) dnevi (rojstni dnevi: 040779, 4/7/79; pomembni dnevi: 20/9/05)
g) povezava z uporabniškim imenom (tomazhiti1)

4.3 Tipi napadov

Razbijanje gesel običajno pomeni razkriti čistopis na podlagi pridobljene oz. ukradene vrednosti zgoščevalne funkcije (v nadaljevanju: hash funkcija). Danes imajo vsi operacijski sistemi gesla shranjena v datoteki. Pri operacijskem sistemu UNIX se ta datoteka nahaja v mapi /etc/shadow. Operacijski sistemi Windows pa imajo v datoteki SAM (Security Account Manager), ki se nahaja v mapi %windir%\system32\config\, shranjeni dve vrednosti enosmernih hash funkcij: LM in NTLM. Podrobnejše informacije o hash funkcijah smo spoznali že v prejšnjem poglavju.

Ko ima napadalec vrednost zgoščevalne funkcije, lahko prične z razbijanja kriptiranega gesla. Sledijo opisi metod razbijanja gesel:

a) *Dictionary attack (napad s slovarjem)* - gre za avtomatsko ugibanje s pomočjo besednega slovarja. Preden beseda vstopi v hash funkcijo se ji doda naključen niz (ti. salt). Rezultat se nato primerja z vrednostjo v datoteki z gesli. Taki slovarji lahko vsebujejo tudi do milijon besed.
b) *Hybrid attack (hibridni napad)* - ta oblika napada je izboljšava zgornje metode. Algoritem vzame besedo iz slovarja, ter zamenja določene črke s posebnimi znaki ali celo doda na konec besede številko. Bolj napredni slovarji uporabljajo tudi besede v več jezikih.

Tipične zamenjave znakov:

- a → @
- i → !, i → 1
- o → 0 (ničla) ali ( )
- S → $, S → 5
- e → €, e → 3
- ...

Primer: pri besedi *password* poskuša naslednje možnosti:

- p@ssword
- pa$$word
- passw0rd
- password1
- ...

c) *Brute force attack (napad z grobo silo)* - to obliko napada se uporablja v primeru, da zgornja dva nista bila uspešna. Namreč ta napad preizkusi vse možne kombinacije znakov do željene dolžine gesla. Toda zavedati se moramo, da metoda grobe sile zahteva veliko procesorskega časa in je zato razmeroma neučinkovita. S programom John The Ripper sem preveril kako se ta metoda obnese v praksi. Zaloga vrednosti znakov šteje 95 ti. printable characters: !"#$%&'()*+,-./0123456789:;<=>?@AB CDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwx yz{|}~ in presledek. Poizkus je tekel na domačem računalniku s procesorjem Intel Pentium Mobile 1.8GHz. Pri tem taktu zmore procesor pregledati 700.000 kombinacij (uporabniško ime skupaj z geslom) na sekundo. Če bi želeli razbiti 4 mestno geslo, moramo preveriti 95^4 = 81.450.625 različnih kombinacij (NTLM avtentikacijska metoda). Časovno to znese skoraj 2 minuti. Z vsakim dodatnim znakom se dolžina slovarja eksponentno poveča. Tako bi za 8 znakov dolgo geslo porabili približno 300 let! Razvidno je da en sam računalnik ni kos tako obsežni nalogi - vsaj ne v doglednem času. Zato je toliko bolj priljubljen porazdeljen napad. Napadalec ročno porazdeli obremenitev iskanja na več računalnikov ali pa to zanj stori program, ki med napadom išče proste vire in porazdeljuje delo.

Sledijo opisi metod kjer se napadalcu sploh ni potrebno ukvarjati z razbijanjem kriptiranega gesla:

a) *Guessing (ugibanje)* - to je najbolj primitivna metoda in sodeč po raziskavah presenetljivo uspešna. Razlog za njeno uspešnost tiči, da uporabniki v veliki večini primerov izberejo prelahka gesla, torej velja najprej poskusiti uganiti geslo.
b) *Social engineering (socialni inženiring)* - s pomočjo manipulacije skušamo pridobiti zaupne podatke od uporabnikov. Kako zelo učinkovita je ta metoda nam pove BBC-jeva raziskava: nekaj več kot 70% uslužbencev v Veliki Britaniji je pripravljeno razkriti svoje službeno geslo v zameno za čokolado!
c) *Phishing (besedna igra besede »fishing«)* - napadalec postavi lažno spletno stran, ki je na videz podobna neki znani strani (npr.: Yahoo! Mail, Gmail, eBay, ...). Kraja uporabniškega imena in gesla se zgodi ob prijavi obiskovalca v lažno prijavno okno.
d) *Shoulder surfing - (»gledanje čez ramo«)* pri tipkanju gesla nam napadalec gleda čez ramo oz. gleda pod prste. Ta metoda je učinkovita kjer je velika množica ljudi, kjer enostavno nimaš nadzora kaj se dogaja za tvojim hrbtom. Priporočljivo je da uporabnik zna geslo slepo natipkati.

4.4 Orodja

Na žalost se najde na spletu ogromno orodij za razbijanje gesel. Ampak če pogledamo iz drugega zornega kota, lahko to vzamemo tudi za srečo, saj lahko vzdrževalci prav tako kot napadalci, pregledujejo varnost računalniškega sistema [2].

Seznam orodij za razbijanje gesel:

a) *John the Ripper* - je odprtokodni program in njegova prednost je razbijanje različnih kriptirnih algoritmom. Prvenstveno je bil napisan za odkivanje šibkih UNIX gesel. Danes pa podpira različne operacijske sisteme (11 UNIX platform, Windos, DOS, BeOS in OpenVMS) [3].

b) *L0phtCrack* (LC5) - je najbolj razširjen program za razbijanje Windows gesel. Orodje pridobi podatke potrebne za razbijanje gesel iz Windows delovnih postaj, mrežnih strežnikov, primarnih kontrolorjev domen in aktivnega imenika. V nekaterih primerih lahko orodje pridobi podatke tudi s pomočjo prisluškovanja omrežja. LC5 omogoča hkratno simultano povezovanje na več računalnikov [4].
c) *Cain & Abel* - zelo zmogljiv programski paket za razbijanje Windows gesel [5].
d) *Crack* - še vedno zelo popularen v UNIX okoljih in velja za standard med programi za razbijanje gesel.
e) *Mio-Star* - je program, pisan za operacijske sisteme UNIX in Linux, ki omogoča porazdeljeno iskanje na računalnikih, ki imajo nameščen Perl.
f) *Saltine Cracker* - razbija tako NTLM hash (MD4) kot POSIX LibDES (CRYPTv3) gesla.

Še vedno so uporabni tudi starejši programi, kot so: *Killer cracker*, *XIT*, *Merlin*, *Viper* in drugi, ki ne omogočajo samodejnega porazdeljenega iskanja, vendar so njihovi algoritmi prav tako uspešni pri razbijanju gesel [4].

Poglavje 5

Izboljšajmo varnost gesel

5.1 Močna gesla

http://www.securitystats.com/tools/password.php

Kaj pravzaprav pomeni pojem močno geslo? Močno oz. robustno geslo je tisto geslo, ki ga ni lahko uganiti. Torej vsak uporabnik bi moral težiti k temu, da bi bilo njegovo geslo za napadalca težko uganljivo [6].

V podpoglavju 4.2 smo navedli primere gesel, katerih se jih moramo izogibati - ti. šibka gesla. Za »izdelavo« močnih gesel pa obstajajo določene dobre prakse in priporočila, te so:

a) Geslo naj ima vsaj 11 znakov. Daljše kot je geslo, težje ga je uganiti.
b) Med črke je priporočljivo pomešati številko/-e. To je veliko boljša izbira kot, če je geslo sestavljeno samo iz črk oz. številk.
c) Geslo naj vsebuje vsaj 3 velike tiskane črke.
d) Geslo naj vsebuje posebne znake: !"#$%&'()*+,-./:;<=>?@[\]^_`{|}~
e) Uporabnik naj zna geslo slepo natipkati. S tem otežimo delo osebi, ki nam »gleda pod prste«.

Tehnike izdelave močnih gesel:

a) *Mnemotehnika* - za lažje memoriziranje gesel se poslužujemo mnemotehnike. Stavek kot je: »Pri šestindvajsetih sem prvič obiskal Los Angeles«, lahko z malo domišljije uporabimo kot močno geslo - »P26sem1.oLA«. Z izbiro slednjega gesla smo zadovoljili prvim štirim zgornjim priporočilom, za petega pa je potrebno še nekaj vaje.

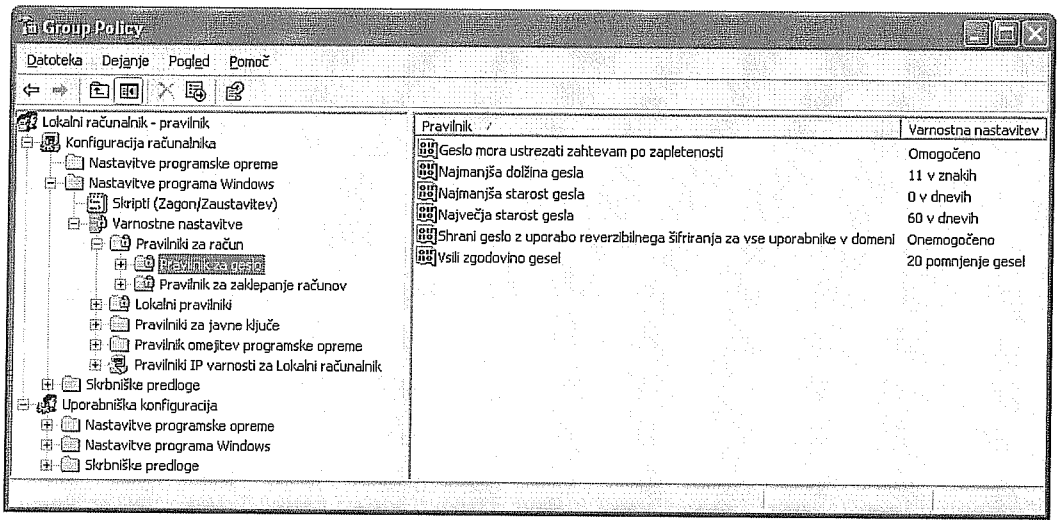
b) *Frazna gesla* - drugi način izdelave močnega gesla je uporaba fraznih gesel (ang. pass phrase). Takšno geslo je običajno veliko daljše kot običajno geslo. Frazna gesla za razliko od običajnih gesel vsebujejo presledke, kar še dodatno izboljša moč gesla. Primer fraznega gesla: »Pri šestindvajsetih sem prvič obiskal Los Angeles«.

5.2 Skrb za gesla

Z izbranim močnim geslom smo šele na začetku poti do izboljšanja varnosti gesel [7]. Naslednja ovira je pomnenje gesla oz. memorizacija gesla. Največja napaka, ki jo uporabniki delajo je zapis gesla na papir. Ta papir je običajno na delovni mizi ali celo prilepljen na monitor! Tudi najmočnejše geslo nam ne pomaga pri takem ravnanju. Zato je edini varen način shranjevanja gesel memorizacija. Pri tem postopku si lahko pomagamo z zgoraj omenjeno tehniko.

Pomembno je tudi menjavanje gesel. S periodiko menjavanja gesel želimo napadalcu otežiti delo. Vzemimo primer da ima napadalec hash vrednost in začne z razbijanjem. Čez čas pa uporabnik spremeni geslo (posledično se spremeni tudi hash vrednost). Torej tudi, če napadalcu uspe razbiti geslo je to neuporabno, saj je bilo geslo medtem spremenjeno.

Pri operacijskih sistemih Windows 2000/XP/2003 je mogoče nastaviti varnostno politiko (ang. security policy) z izdelavo varnoste šablone (ang. security template). V nadaljevanju sledijo priporočila za izdelavo varnoste šablone in razlaga posameznih nastavitev:
Zaženemo gpedit.msc in poiščemo naslednjo politiko:
»Konfiguracija računalnika\Nastavitve programa Windows\Varnostne nastavitve \Pravilniki za račun\Pravilnik za geslo«



Slika 2: Pravilnik za geslo

Razlaga nastavitev varnosti gesla:

- Geslo mora izpolnjevati zahtevo po kompleksnosti: Filtriranje zahtevkov za gesla. Geslo ne sme vsebovati celega ali delov uporabniškega imena. Minimalna dolžina je 6 znakov. Vsebovati mora velike črke, male črke, številke in posebne znake.
- Najmanjša dolžina gesla: Nastavi minimalno število znakov za geslo (1 od 14 znakov). Če nastavimo 0, potem geslo ni potrebno.
- Najmanjša starost gesla: Nastavi minimalni čas veljave gesla, preden ga uporabnik lahko spremeni. Če nastavimo 0 dni, potem lahko uporabnik nemudoma spremeni geslo.
- Največja starost gesla: Nastavi maksimalni čas do poteka gesla (običajno 45 do 90 dni).
- Shrani geslo z uporabo reverzibilnega šifriranja za vse uporabnike v domeni: Nastavi da se geslo shranjuje z uporabo reverzibilnega kriptiranja. To je isto kot, če bi geslo zapisali v TXT datoteko in jo hranili na disku. Zato to pravilo ne bi smelo biti nikoli omogočeno, razen če so zahteve aplikacije pomembnejše od potrebe po zaščiti informacij o geslu.
- Vsili zgodovine gesel: Določa število edinstvenih novih gesel, ki morajo biti povezana z uporabniškim računom, preden je staro geslo mogoče ponovno uporabiti. Vrednost mora biti med 0 in 24 gesli.

5.3 Izbira avtentikacijske metode

Po privzetih nastavitvah vsi Microsoftovi operacijski sistemi podpirajo avtentikacijo preko LM in NTLM. Napadalec vidi v tej privzeti nastavitvi priložnost za vdor. V podpoglavju 3.2 smo prišli do zaključka, da je edina prava izbira avtentikacijske metode NTLMv2. Problem nastane, ko moramo poskrbeti za združljivost za starejše Microsoft operacijske sisteme (Windows 95/98/ME/ NT 4.0). Če želimo v takem informacijskem sistemu omogočati samo NTLMv2 avtentikacijo, moramo najprej onemogočiti LM in NTLM avtentikacijo. To storimo na naslednji način [8]:

1. Za Windows XP/2003
   1.1. S pomočjo Group Policy → Security Options → Network Security: LAN Manager authentication level izberemo »Send NTLMv2 response only\refuse LM & NTLM«
2. Za Windows 2000/NT 4.0 (SP4)
   2.1. Zaženemo urejevalnik registra (regedit.exe).
   2.2. Ključu HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\LSA dodamo zapis »LMCompatibilityLevel« tipa REG_WORD z vrednostjo 5 (krmilniki domen zavračajo odzive protokola LM in NTLM).
   2.3. Ključu HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\LSA\MSV1_0 dodamo zapis »NtlmMinClientSec« tipa REG_WORD z vrednostjo 0x00080000 (varnostna seja protokola NTLM 2)
3. Za Windows 95/98/ME
   3.1. Namestimo Distributed File System (DFS) odjemalca.
   3.2. Microsoft Network Properties nastavimo tako, da vpišemo NetBIOS ime domene.
   3.3. Zaženemo urejevalnik registra.
   3.4. Ključu HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control dodamo zapis »LMCompatibility« tipa REG_WORD z vrednostjo 3 (pošlje samo odziv protokola NTLM 2).

Pred izklop LM in NTLM avtentikacije, se je potrebno temeljito pripraviti na daljše testiranje. Na tem mestu velja še opozoriti, da se stare hash vrednosti nahajajo na diskih do spremembe gesla.

Poglavje 6

Zaključek

Predpostavimo, da se je sistemski inženir držal zgornjih napotkov in tako poskrbel za izboljšanje varnosti informacijskega sistema. Ali lahko inženir še dodatno izboljša varnost gesel? Lahko! Potrebno je poskbeti še za najšibkejši člen te verige - to so uporabniki. Brez ozaveščanja in izobraževanja uporabnikov o problematiki šibkih gesel, lahko naša želja po varnem informacijskem sistemu kaj kmalu postane le mit.

LITERATURA

[1] D. R. Stinson , Cryptography: Theory and Practice, Third Edition, CRC Press Inc., 2005.

[2] Kelley Ealy; A New Evolution in Hack Attacks: A General Overview of Types, Methods, Tools and Prevention; 2003

[3] John the Ripper password cracker; http://www.openwall.com/john/

[4] Stuart McClure, Hacking Exposed: Network Security Secrets & Solutions, Third Edition, Osborne/McGraw-Hill, 2001

[5] Cain & Abel; http://www.oxid.it/cain.html

[6] A. Adams, M. A. Sasse and P. Lunt, Making passwords secure and usable, Paper presented at the HCI, 1997.

[7] Password Security; http://labmice.techtarget.com/security/passwordsec.htm

[8] Protect Against Weak Authentication Protocols and Passwords; http://www.windowsecurity.com/articles/Protect-Weak-Authentication-Protocols-Passwords.html