ORGANIZACIJSKI VIDIKI VAROVANJA INFORMACIJSKIH SISTEMOV (II)

Tomaž Poštuvan

Povzetek:

Prvi del prispevka, ki je bil objavljen v prejšnji številki, je obravnaval varnostna vprašanja kot so zbiranje podatkov, priprava strategije in upoštevanje ekonomskih ter psiholoških vidikov programa varnosti. V pričujočem prispevku pa avtor predstavlja ukrepe za zagotavljanje učinkovitosti programa varnosti, opisuje načine ogrožanja varnosti in rezultate raziskave o varovanju računalniških podatkov v Sloveniji.

Abstract:

The first part of the paper, which has been published in the previous number, discussed security issues such as data collection, development of security strategy and economic and psychologic considerations of security programs. In the present paper, the author presents the necessary actions for security program implementation, describes various security incidents and finally, the results of a research on information security in Slovenia.

■ ■ ■

4. ALI JE PROGRAM VARNOSTI UČINKOVIT?

Vodstvo organizacije mora zagotoviti, da se viri učinkovito izrabljajo. Če so viri tudi zavarovani, potem si mora odgovoriti na naslednji dve vprašanji : (1) Koliko varnosti je dovolj? (2) Ali varnost sploh kaj pomaga? Če se zgradi hiša, so posledice takoj vidne, če se denar troši za varovanje računalniških virov, pa ne. Zato je treba znati oceniti učinek, ki smo ga dosegli s tem, ko smo namestili razne rešitve problema.

Izbira načina zaščite je močno odvisna od osebne sodbe. Ta je lahko profesionalna, brez posebnega poudarka na varovanju podatkov, lahko izhaja iz rezultatov trenutnega stanja v organizaciji ali pa iz analize verjetnosti groženj. Za oceno učinkovitosti programa varnosti je treba :

1. **Določiti, ali program varnosti zadošča ključnim faktorjem uspeha.** To pomeni, da se morajo faktorji izmeriti in primerjati z dejanskim stanjem. Čeprav se lahko okoliščine menjajo, se ključni faktorji uspeha ne smejo.
2. **Določiti obseg že znanih varnostnih incidentov.** Vsako poročilo mora pokazati, kje je največkrat prišlo do incidenta, da se lahko na tisto mesto osredotočimo.
3. **Ugotoviti stopnjo pripravljenosti zaposlenih.** Ker so ljudje glavni nosilci varnosti, je treba vedeti, koliko o varnosti vedo in ali so pripravljeni sodelovati pri izboljšavah.
4. **Napraviti poročilo o primernosti varnostnega programa.** Mnenje o primernosti programa mora vsebovati seznam prejšnjih incidentov in verjetnosti, da se bo incident ponovil na tem ali kakšnem drugem mestu.

4.1 Metode testiranja učinkovitosti

Metode, ki bi povedala, da je program varnosti bodisi dober bodisi zanič, ni. Preizkusi, ki jih bom opisal, so narejeni za oceno vodstva. Posebni preizkusi, ki natančno pregledajo delovanje programa ali prisluškovanje komunikacijam, zahtevajo zelo drago opremo in jih zato ne bom omenjal. Metode testiranja so narejene zato, da se prepričamo, ali program deluje in se imenujejo "90-10 pravila", saj je zato, da preverimo 90 % učinkovitosti potrebnih 10 % naporov, za ostalih 10 % pa 90 % napora. Nesmotrno je torej, da skušamo za vsako ceno doseči 100 % učinkovitost programa.

- **Preizkus "občutka".** To je najenostavnejši preizkus, a včasih tudi najučinkovitejši. Dovolj je, da ugotovimo ali program deluje pravilno ali ne, skozi pogovore z uporabniki, ogledovanjem rezultatov in osebno raziskavo. Močno je odvisen od subjektivne ocene izvajalca preizkusa.
- **Anketiranje uporabnikov.** S tem preizkusom sprašujemo tiste zaposlene, ki imajo opravka z varnostjo ali so za njo odgovorni, o njihovem mnenju ali je program dovolj učinkovit in primeren za organizacijo. Lahko je bolj površen, lahko pa se spusti v podrobnosti, če se ugotovi, da kakšen del programa varnosti ni dovolj zanesljiv.
- **Pregled poročil o varnostnih incidentih.** Večina organizacij ima shranjena pisna poročila o varnostnih incidentih iz preteklosti, čeprav je lahko vsebina teh poročil različna. Namen tega preizkusa je zbrati vse informacije o kršitvah ukrepov varnosti, ki so se zgodili v preteklosti. Na ta način se potem določijo nadaljnji ukrepi.

- **Pogovor z osebo, zadolženo za varnostne incidente.** Važen del programa varnosti je tudi oseba, ki je zadolžena za incidente (tough enforcer). Delavci so zelo nesrečni, ko ugotovijo, da njihov kolega goljufa, vendar ne vedo, komu lahko to povedo, še posebej, če je goljuf nadrejeni. Zato je oseba, ki je za to zadolžena, nujno potrebna. Prijava se lahko napravi tudi anonimno ali pa se za anonimnost jamči, če dotični tako zahteva. Seveda je potreben temeljit pogovor, da se ugotovi, ali je prijava sploh upravičena.
- **Statistika varnostnega sistema.** Varnostni sistemi ponavadi hranijo razne informacije o delovanju sistema, na primer seznam kršitev varnosti, seznam ljudi, ki so imeli dostop do nekega kritičnega mesta, poročila iz sestankov skupine za varnost ...
  Tip informacij je različen od sistema do sistema, skupina za varnost pa mora biti z njimi seznanjena v taki meri, da lahko potegne zaključek o primernosti programa varnosti.
- **Najem strokovnega svetovalca.** Nekateri načini vdora v sistem zahtevajo tako poznavanje samega sistema, da jih niti operativno vodstvo ne pozna. Zato je potrebno najeti zunanjega strokovnega svetovalca, ki izvede določene preizkuse. Za najem se odločimo zato, ker strokovnjak verjetno ve več kot zaposleni, ker je neodvisen od različnih vplivov skupin v organizaciji in ker lahko primerja stopnjo varnosti z varnostjo v ostalih organizacijah ter predlaga izboljšave.
- **Navezava stikov z ostalimi organizacijami.** Praksa je, da se med sorodnimi organizacijami spletejo vezi, po katerih se prenašajo informacije. Informator je oseba, ki ima možnost vpogleda v sistem varnosti, z njim pa naj vzdržuje kontakte vodja varnosti, saj on najbolj pozna stanje v svojem podjetju in bo lahko iz pogovora potegnil kaj koristnega.
- **Primerjava s sorodnimi organizacijami.** Dobro je vedeti, kako se lahko organizacije med seboj primerjajo. Primerjava pokaže ali organizacija namenja varnosti dovolj denarnih sredstev, prav tako pa pokaže tudi tista področja, katerim se namenja premalo pozornosti in so v bistvu šibka točka sistema. Do primerjave lahko pride s posamičnimi obiski, na sestankih skupine organizacij, namenjenim varnosti ali prek svetovalcev (kot je omenjeno v prejšnji točki).
- **Simulacija katastrofe.** Testiranje za primer katastrofe je proces, ki simulira luknjo v sistemu, da bi se pokazalo, ali je program varnosti dovolj dober za odpravo nevarnosti. Trije tipi testiranja so:
  1. **Katastrofa v računalniški sobi.** Ponavadi je napovedana vodstvu sobe, ne pa operaterjem. Shrani in uniči se del kakšnega programa, nato pa zahteva od operaterjev, da s pomočjo varnostnih kopij vzpostavijo staro stanje
  2. **Preverjanje dokumentacije.** Preizkus te vrste je v prvi vrsti namenjen preverjanju ustreznosti dokumentiranih postopkov za primer katastrofe, ne da bi bilo treba katastrofo tudi dejansko simulirati
  3. **Simuliran vdor v sistem.** Skupina, ki izvaja preizkus, resnično želi vdreti v sistem, najprej prek terminala, do katerega nima dostopa, nato izven delovnega časa po telefonu ali s pregledom, če je kakšen terminal ostal prižgan, lahko pa tudi prek fizičnih ovir, da se ugotovi, če so dovolj zanesljive.
     Preizkusi so najučinkovitejši takrat, ko so posamezniki, ki jih izvajajo, seznanjeni z varnostnim sistemom in z najbolj pogostimi metodami vdiranja.
- **Poročilo vodstva.** Odgovornost vodstva je, da so računalniški viri organizacije zadosti varovani, zato morajo zgraditi sistem kontrol in ga dokumentirati v poročilu. Poročilo naj ne bo preveč natančno, temveč naj sprašuje neprijetna vprašanja, ki bodo odgovorno osebo za varnost spodbudila k predlaganju izboljšav.

Ocena učinkovitosti programa varnosti mora dati dve vrsti rezultatov – zbir dejanskih vdorov, šibkih točk sistema in skrbi v zvezi z njimi ter mnenje o primernosti programa.

Dejanske informacije se zberejo na podlagi poročil o vdorih, mnenje o primernosti programa pa je odvisno od skupine, ki ocenjuje njegovo učinkovitost. Ponavadi sloni na tem, kako blizu pride dejansko stanje teoretičnemu stanju, pri katerem bi lahko rekli, da je organizacija varna.

6. RAZLIČNI NAČINI OGROŽANJA SISTEMA IN PROTIUKREPI

Osebje, ki skrbi za varnost v organizaciji, mora informacijski kriminal obravnavati enako kot klasičnega, zato mora biti z metodami ogrožanja sistema dobro seznanjeno. Prav tako je dobro, če ve, kdo ima dovolj znanja, da bi te metode lahko uporabil, ker se tako krog možnih kandidatov močno skrči.

Opisal bom dvanajst metod informacijskega kriminala, v katerem glavno vlogo igrajo računalniki. Pri vsaki bom omenil tudi nekaj načinov, kako metodo odkrijemo in kdo bi jo lahko uporabil.

Prirejanje podatkov *(Data diddling)*

To je najenostavnejša, najvarnejša in najpogosteje uporabljena metoda za računalniški kriminal. Vsebuje popravljanje podatkov pred ali med vnosom v računalnik. Zamenjavo lahko naredi kdorkoli, ki ima dostop do vnosa, shranjevanja, prenašanja ali testiranja podatkov v računalniku. Primeri so ponarejanje dokumentov, zamenjava

trakov ali drugih pomnilniških medijev ter izogibanje običajnim kontrolam vnosa.

Podatki se ponavadi preverjajo ročno, potem ko so enkrat v računalniku, pa pravilnost vnosa preveri še ta. Drugi možen način kontrole vnosa je tudi uporaba dodatne številke ali znaka, vključenega v podatek. Velike množine podatkov se preverjajo s posebej za to napisanimi programi. Dokaz prirejanja podatkov so podatki, katerih dodaten znak ni pravilen, kontrolna vsota ni enaka dodatni številki ali jih program za testiranje izvrže.

Možni kriminalci so zaposleni, ki pripravljajo ali vnašajo podatke ter kdorkoli, ki ima do njih dostop. Tipičen primer prirejanja podatkov je uradnik, ki je nadziral delo tristotih vnašalcev podatkov o delavcih in odkril, da se za obdelavo uporablja le njihova šifra. Tako je napisal nekaj formularjev, v katerih je pod imenom in priimkom drugih delavcev napisal svojo šifro in nadure, ki naj bi jih napravil. Nikoli ga ne bi odkrili, če se slučajno ne bi opazilo, da ima nenormalno visok osebni dohodek za uradnika.

Trojanski konj *(Trojan horse)*

Trojanski konj so na skrivaj dodani ukazi v program, tako da bo računalnik izvedel prepovedane akcije, čeprav ponavadi osnovna funkcija programa ostane nespremenjena - poleg ostalega bo naredil le še nekaj več. To je najobičajnejša metoda pri spreminjanju samega programa. Za odkrivanje ni nobenih pametnih metod, še posebej če je kriminalec dovolj zvit. Trojanski konj je skrit med 5 milijoni vrstic programske kode, kjer čaka na ukaz aplikacije, v nekaj milisekundah doda dodatne ukaze, ki nekaj naredijo in jih ponovno zbriše. Tudi če se dejanje ugotovi, se še vedno ne ve, kdo je to naredil, razen če se med vsemi zaposlenimi, svetovalci in prejšnjimi zaposlenimi poišče tiste, ki imajo za to dovolj znanja.

Trojanski konj se lahko najde tako, da se primerja program z izvirnim programom, za katerega se ve, da se ga ni nihče dotikal. Varnostne kopije niso dovolj, kajti pameten kriminalec bo naredil spremembe tudi tam. Lahko ga najdemo tudi tako, da mu na vhod pripeljemo take podatke, da ga bomo zbudili, čeprav je to v praksi zelo težko izvedljivo. Možni vdiralci so programerji, ki imajo dovolj natančen vpogled v izvorno kodo programa, uporabniki, lahko pa tudi pisci operacijskega sistema, če je bil seveda sistem napisan le za to organizacijo.

Tehnika majhnih zalogajev (Salami technique)

Tehnika majhnih zalogajev se tako imenuje zaradi tega, ker se vzame le majhen kos naenkrat, tako da se kraje sploh ne opazi. Na primer, v banki uslužbenec s pomočjo Trojanskega konja iz računa vsakega varčevalca pri obračunu obresti (ker se seveda zneski zaokrožujejo) vzame znesek od tretje decimalke naprej, ker pa je varčevalcev več deset tisoč, se v enem letu tega denarja nabere kar precej. Uspeh sloni na tem, da nihče ne bo opazil, da se mu je znesek zaokrožil (metoda zaokrožanja navzdol).

Nadzornik lahko krajo odkrije samo na dva načina: ali ročno pregleda vse ukaze programa ali pa na roko izračuna znesek obresti. Pameten programer lahko Trojanskega konja skrije tako dobro, da se ga brez natančnega pregleda ne bo dalo odkriti, vendar pa to verjetno niti ne bo potrebno, saj nadzornik ne bo pregledoval programa, dokler bo deloval "pravilno". Vdiranje se lahko odkrije predvsem tako, da se pazljivo spremlja finančno stanje osumljencev, predvsem zaposlenih. To prisili kriminalce, da odprejo račun pod geslom, kar pa je že znak, da je nekaj narobe in da je treba račun spremljati še bolj pazljivo.

Možni vdiralci so programerji finančnega sistema, zaposleni ali prejšnji zaposleni.

Superzap *(Superzapping)*

Superzap je program, ki se uporablja v večini velikih računalnikov podjetja IBM kot sistemsko orodje. Vsak računalniški center potrebuje orodje kot je Superzap, da lahko obide običajne varnostne poti v primeru zrušitve sistema.

Orodja take vrste so zelo nevarno orodje v napačnih rokah. Običajno so na uporabo le sistemskim programerjem in vzdrževalcem informacijskih sistemov, včasih pa se vseeno nahajajo v knjižnicah, kjer so na razpolago vsakomur, ki bi jih znal uporabljati.

Klasičen primer Superzapa je sistemski inženir, ki je program uporabljal v banki za popravljanje zneskov na računih po ukazu vodstva banke. Ko je videl, kako enostavno je popravljati zneske, je malce popravil tudi zneske svojih treh prijateljev. Ujeli so jih le zato, ker so to delali dovolj dolgo, da se je ena od strank pritožila.

Dejanja Superzapa je možno odkriti tako, da se podatkovna baza primerja s svojo kopijo in prejšnjo kopijo, na katerih se program še ni uporabil. Kriminalci so lahko programerji, ki imajo dostop do programa, ali tisti zaposleni, ki imajo za uporabo programa dovolj znanja.

Skriti pristopi *(Trapdoor)*

Pri razvoju velikih aplikacij je navada programerjev, da se v kodo dodajo deli programa, preko katerih se lahko dodajajo nove funkcije ali pa se uporabljajo za ogledovanje (debugging). Ti deli programa se imenujejo skriti pristopi (traps). Normalno je, da se izločijo iz programa, preden se naredi končna verzija, včasih pa se jih spregleda ali pa namerno pusti zaradi kasnejšega vzdrževanja. Lahko pa se uporabijo tudi v slabe namene. Možni krivci so sistemski ali navadni programerji z veliko znanja.

V nekem primeru računalniškega kriminala je sis-

temski programer odkril možnost skritega pristopa v prevajalniku jezika FORTRAN. Pristop je omogočal prenos podatkov iz programa v področje, ki se uporablja za shranjevanje podatkov. Na ta način je lahko s programom mimo kontrol vnašal svoje podatke v bazo in se s tem okoristil.

Za odkrivanje nastavljenih skritih pristopov ni kakšne posebne metode. V primeru suma pa se vseeno lahko napravijo določeni preizkusi, ki odkrijejo skrite funkcije v programu, čeprav to precej stane (porabi se zelo veliko časa, potrebni so strokovnjaki).

Logične bombe *(Logic bombs)*
Logična bomba je program, ki se izvaja ob določeni uri (lahko tudi periodično) v sistemu. Programirana je tako, da lahko sproži akcijo ob kakršnemkoli pogoju. V programsko kodo se dodaja kot Trojanski konj, tako da so tudi metode za odkrivanje podobne kot pri odkrivanju Trojanskih konjev. Kriminalci so lahko programerji z dovolj znanja, zaposleni, prejšnji zaposleni in tudi uporabniki programov.

Primer logične bombe je programer, ki je v program dodal Trojanskega konja, ki bo čez dve leti ob določenem času na vseh terminalih izpisal priznanje krivde, nato pa zrušil sistem. Tempirano je bilo tako, da je bil programer v tem času že daleč stran.

Časovno neodvisni napadi *(Asynchronous attacks)*
Časovno neodvisni napadi slonijo na neusklajenosti delovanja operacijskega sistema. Primer neusklajenosti je, ko več programov istočasno zahteva izpis na tiskalnik. Operacijski sistem shrani zahteve v vrsto, iz katere nato tiskalnik jemlje zahteve po neki prioriteti. Pri tem pa lahko pride tudi do poskusa napada. Vzemimo program, ki obdeluje ogromno število podatkov. V program morajo biti vgrajene točke, v katerih lahko programer delovanje prekine in nadaljuje, denimo, naslednji dan. Zato mora biti operacijski sistem zmožen shraniti kopijo programa in trenutne podatke, prav tako kot številne sistemske parametre. Če programer dobi dostop do kopije programa ali podatkov, potem lahko spremeni parametre tako, da bo imel program višjo prioriteto in s tem dostop do ostalih podatkov ali celo operacijskega sistema.

Posledice kriminalnega dejanja postanejo vidne šele ob čudnem obnašanju aplikacije ali sistema, možni krivci pa so predvsem sistemski programerji, kajti navadni programerji do kopij programa nimajo dostopa.

Stikanje po odpadkih *(Scavenging)*
Stikanje po odpadkih je pridobivanje informacij, ki ostanejo v sistemu po končanju naloge. Enostavno stikanje je iskanje po koših za smeti za računalniškimi izpisi, težje pa je iskanje po računalniškem pomnilniku ali disku. Primer je premalo natančno brisanje, po katerem vsebina še vedno ostane na disku, čeprav je operacijskemu sistemu nevidna. Do odkritja uporabe stikanja pride šele takrat, ko pride do zločina, za katerega so bile potrebne informacije, ki so lahko prišle samo iz računalnika.

Primer stikanja po odpadkih je uporabnik, ki je vedno pred začetkom dela zahteval, da se v računalnik postavi nov magnetni trak. Operaterju se je to zdelo čudno, še bolj čudno pa se mu je zdelo zato, ker je vsakič, preden je uporabnik nanj karkoli zapisal, začela goreti lučka za branje. Odkrilo se je, da je uporabnik najprej prebral ne dovolj dobro zbrisane informacije iz traku in jih prodajal konkurenčnim podjetjem.

Odtekanje podatkov *(Data leakage)*
Praktično pri vseh primerih računalniškega kriminala gre za prilastitev podatkov ali programov iz sistema. Za odtekanje podatkov iz sistema obstaja več poti - ena je, da se podatki skrijejo med ostale podatke v poročilu, izpisanemu na tiskalnik. Še bolj zapletena je, če se podatki zakodirajo tako, da izgledajo drugačni kot v resnici (različno število znakov v vrstici, število besed v vrstici). Obstajajo sicer še bolj eksotične poti, kot na primer opazovanja gibanja magnetne glave traku, zvok tiskalnika, posnet na kaseto itd., vendar so malo verjetne in pridejo vpoštev le v dobro varovanih sistemih.

Primer odtekanja se je pripetil ameriški vojski v vojni z Vietnamom. V računalnikih v Vietnamu so bili skriti majhni radijski oddajniki, ki so oddajali njihove podatke oddaljenemu sprejemniku. Seveda so to odkrili šele potem, ko je bila vojna v Vietnamu že končana.

Odtekanje se vodi s Trojanskimi konji, logičnimi bombami ali stikanjem, odkrije pa se najlažje z zasliševanjem tistega osebja, ki bi ga morali opaziti ali pa s preverjanjem zadnjega dostopa do datoteke. Dokazi, da je prišlo do odtekanja , so praviloma enaki kot pri stikanju.

Tihotapljenje in napačno predstavljanje *(Piggybacking and impersonation)*
Tihotapljenje in napačno predstavljanje se lahko pojavita v dveh oblikah, fizični in elektronski. Do fizičnega tihotapljenja pride takrat, ko se kontrola vrši pri elektronsko ali mehanično zaprtih vratih. Tipično gre zaposleni s polnimi rokami računalniških stvari skozi vrata (pri tem uporabi ključ ali identifikacijsko kartico), goljuf pa se izmuzne skozi odprta vrata.

Za preprečevanje takih primerov je potreben varnostnik ali zapora, ki prepušča naenkrat samo eno osebo (imajo jo v veleblagovnicah, na podzemskih železnicah ...).

Elektronsko tihotapljenje se po drugi strani uporablja za dostop do računalnika - na isto linijo s terminalom se priključi še en terminal in uporablja takrat, ko avtorizirani uporabnik svojega ne uporablja. Do elektronskega tihotapljenja lahko pride tudi v primeru, ko uporabnik, potem ko je vnesel svoje geslo, pusti računalnik prižgan in s tem na voljo mimoidočim.

Napačno predstavljanje je proces, ko se oseba predstavi kot nekdo drug. Kot protiukrep napačnemu predstavljanju velja identifikacija na podlagi prstnih odtisov (geslo v računalniku je neuporabno, saj ga lahko na tak ali drugačen način odkrijejo) ali glasu.

Nadzor nad napačnim predstavljanjem se vrši s pregledovanjem zapisnikov, ki se vodijo ob vsakem prihodu in odhodu in pogovori z ljudmi, ki so bili priča neavtoriziranemu dostopu.

Primer napačnega predstavljanja je človek, ki je klical v podjetje in se predstavil kot novinar, ki piše članek o računalniškem sistemu, ki ga v podjetju uporabljajo. Potem ko so mu v podjetju natančno razložili delovanje sistema, je bil zmožen ukrasti opremo, vredno več kot milijon ameriških dolarjev.

Prisluškovanje *(Wiretapping)*

Prisluškovanje je klasična metoda pri komunikacijah na daljavo, saj ga s pravo (sicer drago) opremo lahko izvajamo zelo enostavno. Problem je le v tem, ker se nikoli ne ve, kdaj se bodo pošiljali zanimivi podatki, zaradi tega se mora podatkov zbrati ogromno. Zato je prisluškovanje najmanj verjetna metoda informacijskega kriminala.

Najučinkovitejši protiukrep prisluškovanju je šifriranje podatkov - trenutno obstaja kar nekaj šifer, ki se danes pojmujejo kot nezlomljive, saj imajo prek 10^{98} različnih kombinacij.

Modeliranje in simulacija *(Modeling and simulation)*

Računalnik se lahko uporabi tudi kot orodje za planiranje in kontrolo kriminalnih dejanj. Predvideno dejanje se simulira na računalniku in se tako oceni, kakšna je verjetnost, da bo uspelo. V primeru zavarovalne tatvine je računalnik izračunal, kakšne bodo posledice, če bi prodali veliko število zavarovalnih polic. Rezultat modeliranja je bil nastanek 65.000 ponarejenih zavarovalnih polic, prodanih zavarovalnicam.

Uporaba modeliranja in simulacije zahteva zelo veliko procesorskega časa in razvoja, zato so osumljenci predvsem tisti, ki so ga porabili največ. Ostali možni krivci so še strokovnjaki za modeliranje in simulacije ter programerji.

7. VAROVANJE RAČUNALNIŠKIH SISTEMOV V SLOVENIJI

Predstavil bom rezultate analize, ki je bila napravljena leta 1992 in je zajela 250 največjih slovenskih podjetij. Sredstva, ki jih podjetja namenjajo varnosti podatkov se zaradi relativne majhnosti na morejo primerjati s sredstvi, ki jih v ta namen namenjajo podjetja razvitih zahodnih držav. Izpolnjene vprašalnike je poslalo le 50 % anketiranih podjetij, kar lepo kaže, koliko pozornosti varnosti posvečajo podjetja iz Slovenije. Tudi od teh jih več kot 67 % priznava, da varnosti ne posveča dovolj pozornosti, 58,9 % pa namerava v varnost vlagati več, kot je letna stopnja inflacije.

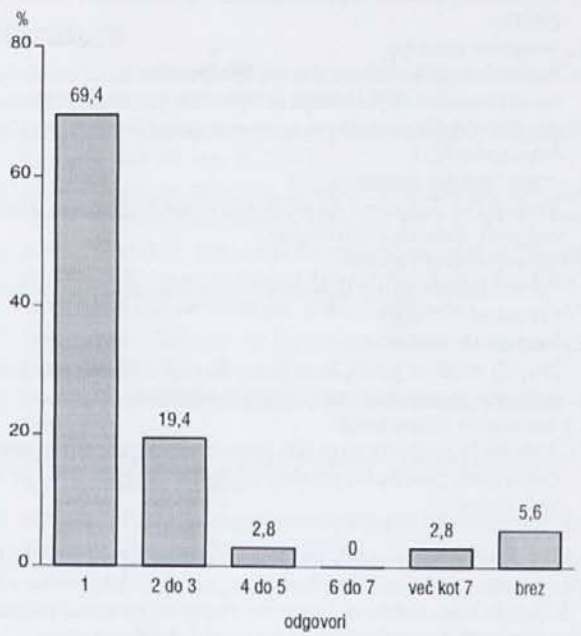
Oseb, odgovornih za varnost podatkov, je v teh podjetjih praviloma zelo malo, največkrat celo samo eden (slika 3), iz česar lahko sklepamo, da odgovornost v večini podjetij še ni jasno opredeljena. Mogoče je neke vrste opravičilo v tem, da jih ima nad 80 % manj kot 2000 zaposlenih, ki za delo v glavnem uporabljajo samostojne mikroračunalnike.

V zadnjih dveh letih se je slika močno spremenila, saj so vsi mikroračunalniki na tak ali drugačen način povezani v omrežje in zato bolj ranljivi, vendar žal v zadnjem času ni bilo nobene podobne ankete, s katero bi lahko primerjali podatke.

Strategijo varnosti, ki je temelj, iz katerega izhaja vse ostalo, ima razdelano le 54,8% podjetij iz testnega vzorca (za primerjavo naj povem, da je delež v zahodnih državah po anketi COMPSEC iz leta 1991 precej višji - 72 % ([2, stran 76]). Od tistih, ki strategije še nimajo, razmišlja o njeni potrebi približno dve tretjini podjetij. Bolj zaskrbljujoč od tega je podatek, da kar 76,7 % vodstvenih delavcev in 78,1 % ostalega osebja problemov z varnostjo podatkov ne jemlje resno.

Žrtev računalniških zlorab je bilo 11 % anketiranih podjetij, od katerih pa so jih le desetino povzročili s zaposleni. Nobeno od podjetij ni kazensko preganjalo storilcev, ker niso povzročili večje škode. Sodeč po anketi se v podjetijh precej bolj bojijo naravnih katastrof, poplav, sabotaž in računalniških virusov.

Rezultati ankete kažejo, da se Slovenci sicer zavedamo problema varnosti podatkov, vendar mu še ne posvečamo dovolj pozornosti. Eden od razlogov je pomanjkanje



Slika 3 **Število zaposlenih, ki so neposredno odgovorni za varnost podatkov**

usposobljenega osebja, drugi razlog pa je, da storilci še niso povzročili večje škode. Ko bo do nje prišlo, bo že malce pozno.

Več o anketi si lahko preberete v [3], napravljena pa je bila še ena anketa, v katero so bila vključena podjetja iz avtomobilske industrije, gradbeništva, zdravstvene ustanove ter trgovska podjetja, ki uporabljajo elektronsko izmenjavo podatkov (EDI - Electronic Data Interchange). Rezultati te ankete pa so natančno opisani v [4].

■

ZAHVALA

Zahvaljujem se prof. dr. Boštjanu Vilfanu za vse nasvete in smernice, ki mi jih je dal in so pomagale pri pripravi tega članka.

■

DODATEK A - Razredi varnosti

Varnost informacijskega sistema mora biti zadolžitev vseh zaposlenih. Deluje le v primeru, če jo tako vodja varnosti kot vsi člani podjetja vzamejo kot osebno odgovornost.

Kriteriji, ki posamezne sisteme varnosti združujejo v skupine, so razdeljeni na štiri razdelke, A, B, C in D, pri čemer najvišjo stopnjo varnosti nudijo sistemi, ki spadajo v skupino A in so tako varni, da jih lahko uporablja Ministrstvo za obrambo Združenih Držav Amerike, NASA itd. (Unix na primer spada v skupino C). Sistemi, ki bi imeli stopnjo varnosti višjo kot A trenutno še ne obstajajo.

Po specifikaciji TCSEC (Trusted Computer System Evaluation Criteria) obstaja sedem razredov, za katere velja, da izhajajo iz svojih predhodnikov - npr. razred B1 vključuje poleg svojih še vse zahteve razreda C2.

D - minimalna zaščita
Malo ali nič varnostnih ukrepov (MS-DOS)

C1 - zasebnost uporabnikov
Nekatere oblike kontrole dostopov; omejen dostop do zasebnih podatkov posameznih uporabnikov (osnovni UNIX)

C2 - kontrola dostopa
Natančnejša kontrola dostopa; uporabniki se prijavljajo na računalnik, kjer imajo svoj račun (account). Dostopi do določenih podatkov se sproti beležijo (UNIX, VMS, Windows NT)

B1 - označevanje podatkov
Podatki so označeni na različnih nivojih (zaupno, strogo zaupno, državna skrivnost)

B2 - strukturirana zaščita
Jasno določena in dokumentirana strategija varnosti (Trusted XENIX)

B3 - varnostne domene
Dovolj majhna koda, ki se je ne da razbiti. Lahko jo preizkušamo in analiziramo v smislu varnosti

A1 - varnostno preverjanje
Sistemi te vrste morajo biti preverjeni z različnimi matematičnimi postopki, preden jih lahko proglasimo za dovolj varne

Več o razredih varnosti z obširnim opisov kriterijev, ki jim mora posamezen razred zadostavati, si lahko preberete v [1].

Kljub še tako dobremu sistemu varnosti pa moto računalniške varnosti še vedno spada na vsako delovno mizo:

Računalniška varnost - če jo hočemo imeti, je odvisna od mene samega.

DODATEK B

Nekaj možnih šibkih točk informacijskih sistemov

V dodatku bom naštel še nekaj šibkih točk informacijskih sistemov, ki so tipične za običajno organizacijo. Seznam niti slučajno ni popoln, vendar lahko spomni na kakšno podobnost in predlaga rešitev:

Napačen vnos podatkov

Napačen vnos podatkov je najobičajnejši način, zaradi katerega pride do napačnega delovanja sistema. Ranljivost je povsod, kjer se podatki zajemajo, obdelujejo ali pripravljajo na vnos v računalnik :

- nedosleden vnos podatkov je ostal neopažen
- nepopolni zapisi so obravnavani normalno, čeprav manjkajo pomembni podatki
- zaposleni lahko namerno dodaja, briše ali spreminja podatke za lastno korist
- zaradi pomanjkanja kontrole vnosa podatkov se lahko transakcije izgubijo ali neopaženo dodajo
- podatki, ki pridejo zadnjo minuto zaradi časovne stiske niso pravilno preverjeni

Zloraba pravic avtoriziranih uporabnikov

Uporabniki so ljudje, zaradi katerih sistem sploh obstaja. Napravljen je za njihovo uporabo, vendar ga lahko zlorabijo za nečedne posle. Pogosto je zelo težko ugotoviti, če so njihove pravice v skladu z delom, ki ga opravljajo :

- zaposleni lahko proda strogo zaupne podatke nekomu, ki mu veliko pomenijo - zavarovalnici, konkurenčnemu podjetju ...
- nekontrolirano se lahko spremenijo podatki o zaposlenemu, ki sicer do njih nima pravice dostopa
- odpuščeni delavec lahko uniči podatke na tak način, da so neuporabni in jih ni niti na varnostnih kopijah
- avtorizirani uporabnik lahko sprejme podkupnino in spremeni določene podatke

Nezavarovan dostop do sistema

Organizacije se izpostavljajo nepotrebnemu riziku, če nimajo fizične kontrole dostopa do računalnikov :

- podatki in programi so lahko ukradeni
- posameznikov ne kontrolirajo dovolj natančno
- oddaljeni terminali niso zaščiteni pred uporabo neavtoriziranih uporabnikov
- uporabnik brez pravice dostopa pride do sistema po telefonski liniji
- uporabniki gesla pišejo kar na rob računalnika ali pa se ga dobi tako, da se mu gleda preko hrbta
- odpuščeni delavec ima dostop do sistema, ker ga niso takoj zbrisali iz seznama zaposlenih

Postopkovne napake

Tako napake kot namerna dejanja lahko pripeljejo do nepravilnih postopkov, spodrsljajev pri kontroli in izgube podatkov :

- podatki se lahko uničijo med brisanjem diska ali reorganizacijo podatkovnih baz
- nepravilen zagon sistema vodi do izgube posameznih transakcij
- izvaja se napačna verzija programa
- program se izvaja na napačnih podatkih
- ključni pomnilniški mediji (trakovi, diski) se uporabljajo, ne da bi bili zaščiteni pred pisanjem
- začasni trakovi niso dovolj dobro zbrisani po uporabi
- izhod operacije je poslan na napačen terminal

Programske napake

Aplikacijski programi naj bodo razviti v okolju, ki zahteva popoln, pravilen in natančen razvoj, primerno testiranje, dobro dokumentacijo in postopke za vzdrževanje.

Čeprav bodo programi, razviti v takem okolju, še vedno imeli napake, jih bo precej manj in bodo lažje odkrite. Prav tako programerji programa ne bodo mogli "popravljati" (dodajati npr. Trojanskih konjev), saj se potem ne bodo več ujemali s specifikacijami :

- zapisi važnih datotek se lahko pobrišejo, ne da se bi jih dalo dobiti nazaj iz varnostnih kopij
- programerji lahko dodajajo svojo kodo v program
- spremembe programa niso dovolj testirane pred uporabo
- testiranje ne odkrije napake, ki se pojavi le ob določeni kombinaciji tipk
- dokumentacija ni dovolj dobro varovana
- zaposleni ukrade program in ga uporablja za svojo lastno uporabo
- napake nastanejo, ker programer ni dobro razumel, kakšne spremembe naj napravi v programu

Napake operacijskega sistema

Napake v operacijskemu sistemu in namerni vdori z namenom popravkov operacijskega sistema lahko povzročijo večje probleme kot aplikacije, poleg tega pa se težje odkrijejo:

- uporabnik lahko bere ali piše izven področja, ki mu je dodeljen
- zaradi zrušitve operacijskega sistema se lahko dobijo podatki o geslih ali pravicah dostopa
- vzdrževalno osebje se ne kontrolira na vhodu, tako da je v bistvu v računalniškem centru neevidentirano in lahko to izkoristi za kakšno podlo dejanje
- pri ponovnem zagonu sistema po zrušitvi sistemu ne uspe zagotoviti, da se za terminali še vedno nahajajo iste osebe kot pred zrušitvijo
- uporabnik lahko pride v nadzorni ali sistemski način delovanja brez dovoljenja.

Komunikacijski sistem

Podatki, ki potujejo po komunikacijskih linijah, so občutljivi na fizično prekinitev povezave, na prisluškovanje in na spremembe naslovnika, ki jih naredijo neavtorizirane osebe, npr.

- neopažene napake v komunikaciji pomenijo napačne podatke na sprejemni strani
- podatki so lahko preusmerjeni na napačen terminal
- komunikacijskim linijam se lahko prisluškuje
- neavtoriziran uporabnik se polasti komunikacijskih vrat, potem ko jih avtoriziran preneha uporabljati
- če se uporablja šifriranje, se ključ lahko ukrade
- sporočila se lahko posnamejo in ponovno predvajajo (npr. polog 10.000 SIT)

Nekaj naslovov na Internetu

Programska oprema za varnost podatkov :

- Auditor – program za preprečevanje kraj programske opreme
- Centri (http://www.cohesive.com/centri/what.htm)
  - programska rešitev, ki omogoča centraliziran dostop do Interneta, varnost pretoka podatkov in preprečevanje varnostnih incidentov
- Betsi (http://info.bellcore.com/BETSI/ general.info.html)
  - prost program (freeware), s pomočjo katerega lahko proizvajalci programske opreme razširjajo svoje programe in njihove popravke kar prek Interneta; za identifikacijo uporablja PGP (Pretty Good Privacy) javne ključe
- PC Security Ltd. (http://usa.net/pcsl/prdinfo.html)
  - podjetje, ki je razvilo več produktov za varnost, od računalniških sistemov pa do elektronske pošte (StopLock)
- The Federated Software Group Inc. (http://www.federated.com)
  - podjetje, ki se prav tako ukvarja z razvojem programov za varnost podatkov, predvsem pri podatkovnih bazah
- http://www.yahoo.com/Business/Corporations/Computers/Security
  - informacije o podjetjih, ki se ukvarjajo s prodajo programske opreme za varnost informacijskih sistemov

Dodatne informacije o varnosti:

- http://www.yahoo.com/Science/Mathematics/Security and Encryption
  - informacije o kriptografiji, vezju Clipper, računalniških virusih, požarnih zidovih (firewalls) ...
- http://www.cohesive.com/secure.htm
  - tehnologije postavitve požarnih zidov
- http://www.ascinet.com/safeware/index.html
  - izbira načina strategije varnosti za organizacijo
- http://rainier.cs.ucdavis.edu/Security.html
  - raziskovalni laboratotij univerze v Davisu, Kalifornija, ki se ukvarja z odkrivanjem vdorov v sisteme, razvojem varnih protokolov prenosa podatkov itd.
- http://www.sei.cmu.edu/tech/compusec.html
  - zelo dobra referenčna točka za nadaljnje iskanje, denimo do koordinacijskega centra za pomoč uporabnikom Interneta CERT, ki se ukvarja z odkrivanjem in preprečevanjem vdorov v sisteme.

Literatura

[1] Department of Defense: *Department of Defense Trusted Computer System Evaluation Criteria*, 1985

[2] Paul Evans: *Conference Report; COMPSEC 90*, Computer & Security, Vol 10, No. 1, 1991

[3] Alenka Hudoklin, Branislav Šmitek: *Computer Systems Security in Slovenia*, Computer & Security, Vol 13, No. 1, 1994

[4] Alenka Hudoklin, Branislav Šmitek: *Assesment of Organization Security Level Before EDI Implementation*, The Fifth International EDI Conference, Bled, 1992

[5] D.W. Davies: *Security for Computer Networks*, John Wiley & Sons, 1984

[6] Paul J. Fortier: *Handbook of LAN Technology*, McGraw-Hill, 1992

[7] Roger M. Needham: *Denial of service*, Communications of the ACM, November 1994

[8] William E. Perry: *Management Strategies for Computer Security*, Butterworth Publishers, 1985

◆

Tomaž Poštuvan je diplomiral leta 1993, trenutno pa je zaposlen kot mladi raziskovalec na Fakulteti za računalništvo in informatiko. Njegovo delovno področje so prevajalniki (v tem okviru tudi pripravlja magistrsko delo), sicer pa se je precej ukvarjal tudi z varnostjo oz. zaupnostjo podatkov v računalniških sistemih.