Semestrální práce Y36SPS

Generátor zabezpečovacího skriptu

Filip Hašek **2008/2009**

Zadání:

Vytvořit skript, který na základě vstupu uživatele vygeneruje skript pro zabezpečení iptables a jádro systému.

Postup:

Jako první část jsem si vytvořil seznam toho, co lze zabezpečit. Nejdříve jsem hledal tzv. best practices a podobná klíčová slova. Z analýzy vyšlo, že naprostá většina možnosti zabezpečení jsou buďto iptables a nebo konfigurační soubory jednotlivých služeb. Vzhledem k mechanismu jak jsou v Linuxu ukládány konfigurace, tedy textové navzájem includované soubory, jsem došel k závěru, že vytvářet funkce pro hledání jednotlivých konfiguračních direktiv by zabralo příliš času vzhledem k užitečnosti. Tato část úkolu byla tedy vynechána v prospěch iptables a síťových parametrů pro jádro.

Pokračoval jsem analýzou jednotlivých síťových služeb a zabezpečení via iptables. Šlo především o hledání nejčastějších služeb a k nim specifikace portů a protokolů. Hledání probíhalo na základě vlastní zkušenosti s provozováním OS Linuxu jako serveru i pracovní stanice.

Po vytvoření seznamu nutných služeb jsem je rozdělil na skupiny, které spolu souvisely dle rolí, tedy PC, server a router. K těmto skupinám ještě přibyla obecná nastavení, která se hodí ke všem. Tyto skupiny jsem poté rozdělil na jednotlivé služby. Každé službě bylo vytvořeno nastavení dle výchozího čísla portu.

Součásti:

Prostředí: OS Linux, Fedora 10: Byla vybráno pro svoji univerzálnost. Jelikož je podporována společností RedHat, vychází z ní několik jiných distribucí.

Jazyk: Bash: tento skriptovací jazyk jsem zvolil kvůli jeho snadné dosažitelnosti. Ačkoliv například C či Java jsou schopny poskytnout lepší model, dokonce i objektově orientovaný, je nutno je kompilovat či provozovat prostředí pro ně. Bash je dostupný jako výchozí nástroj ve většině distribucí.

Struktura:

Jde o skript pro bash. Na začátku skriptu jsou proměnné pro specifikaci k potřebným binárním souborům, tj.: iptables a cílový vygenerovaný skript. Další část skriptu obsahuje jednotlivé funkce, které tvoří pravidla pro iptables. Jsou pojmenovány anglicky, přičemž je

snaha, aby byly tématicky blízko sebe. Žádná funkce pro nastavení iptables se nevolá s parametrem, krom pár funkcí určených pro nadřazené volání. Například:

```
function allowOutgoingFTP {
 echo "/sbin/modprobe ip_nat_ftp" >> $SOUBOR
 echo "/sbin/modprobe ip_conntrack_ftp" >> $SOUBOR
 echo "$IPTAB -A OUTPUT -p tcp -m multiport --dport 20,21 -j ACCEPT"
>> $SOUBOR
```

Druhá část představuje interaktivní dialog pro specifikace uživatelových potřeb. V závislosti na uživatelových odpovědích volá skript jednotlivé funkce, např:

```
echo "Povolit veskerou komunikaci na loopbacku? (a/n)"
read souhlas
if [ $souhlas == "a" ]
then allowLoopback
fi
```

Obecná pravidla jsou například funkce jménem:
loadDefaultModules
clearIptables
setDefaultPolicy
invalidPackets

Skript je možno využít krok po kroku, tedy uživatel zodpovídá otýky na ano/ne či vyplňuje hodnoty. Jiné možné volání skriptu je přes název skriptu s argumentem, který je tvořen touto syntaxí:

skript limit-invalid 80,out 443,out, 5190,out

Takto zavolaný skript omezí nevalidní pakety a umožní průchod odchozí komunikaci na portech 80,443 a 5190, přičemž pakety z druhé strany jsou přijmuty.

Seznam všech platných příkazů je k dispozici přes –help.

Skript po svém ukončení vygeneruje soubor do adresáře /etc/init.d/, změní mu vlastníka na root a dá práva 750. Následně vytvoří službu (pokud již není), se jménem skriptu ze kterého byl volán. Na vygenerovaný soubor je možno uplatnit příkazy start a stop.