CSIRT.SK

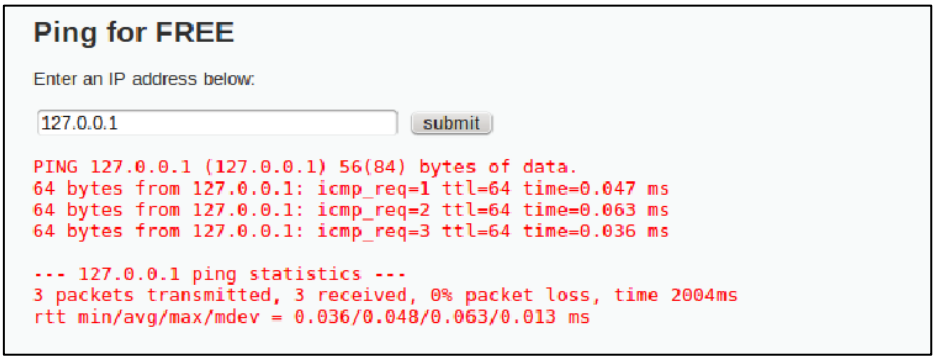
Základná ochrana pred útokmi na web I

Odporúčanie VJ CSIRT

08.04.2024

1. Neoprávnené vykonávanie príkazov

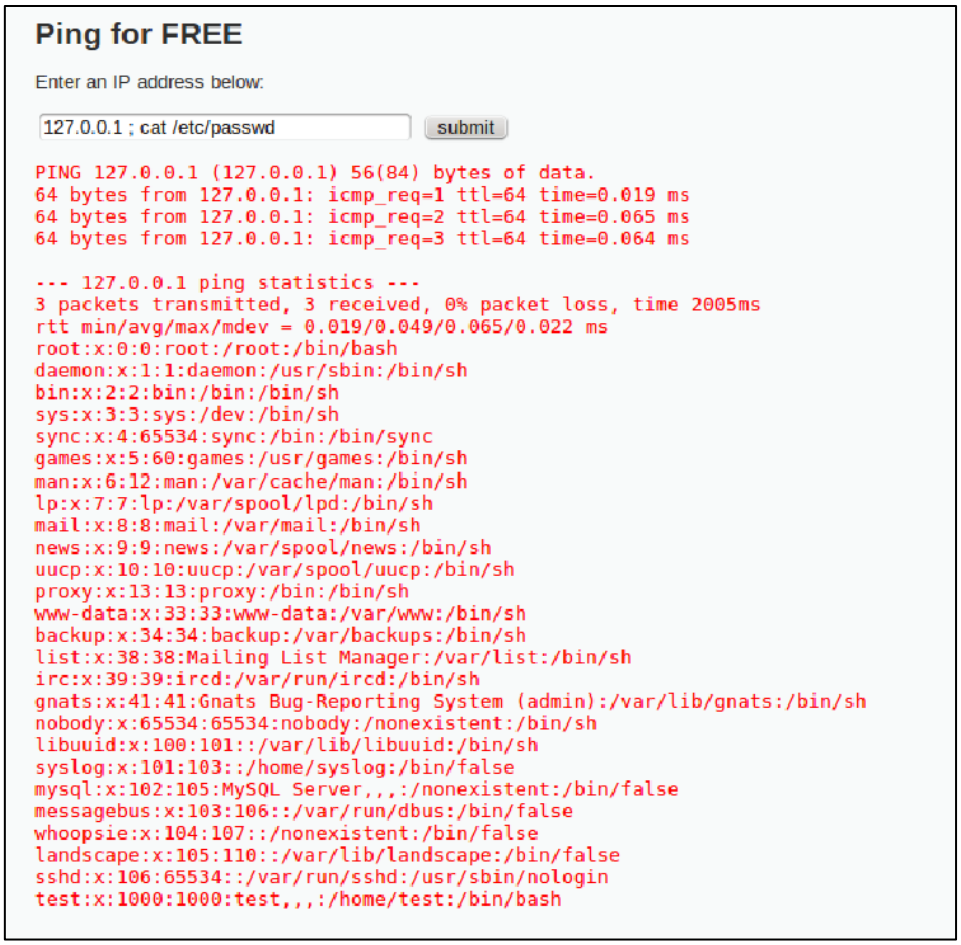
Pod neoprávneným vykonávaním príkazov rozumieme zneužitie systému na vykonanie príkazu, ktorý nebol mienený na vykonanie v prvom rade. Ako príklad uvedieme možnosť vykonať príkaz ping, pre ktorý stačí zadať IP adresu cieľa.



Vďaka nedostatočnej implementácii je ale možné zneužiť mechanizmus, ktorým sa služba na serveri spúšťa.

Útočník dokáže využiť funkciu príkazového riadku na zreťazenie príkazov, a teda vykonať viacero príkazov súčasne. Útočník takto dokáže vykonávať akékoľvek príkazy s právami používateľa, ktorý spustil web server. Môžeme použiť operátor „**;**“, „&&“ alebo „**|**“ na zreťazenie príkazov a vytvoriť tak napríklad príkaz „**127.0.0.1 ; cat /etc/passwd**“, kde sa najprv vykoná prvá legitímna časť a následne kód útočníka, ktorý zobrazí obsah súboru /etc/passwd. Na obrazovku sa vypíše celkový výstup procesu.

Bez ošetrenia takejto zraniteľnosti otvárame veľmi vážnu dieru do systému, kde aj útočník na veľmi nízkej technickej úrovni dokáže spôsobiť rozsiahle škody.



1.1. Ochrana

Ochrana pred takýmto typom útoku spočíva v zablokovaní vkladania iných reťazcov ako tých, ktoré sú očakávaným vstupom. V našom prípade funkcie PING očakávame vstup vo formáte IP adresy **X.X.X.X**, kde X musia byť čísla o max. veľkosti jedného bajtu. Číslo spracúvame po jednotlivých zložkách adresy, a teda sa nám nikdy nevykoná prípadný vložený kód. Čím detailnejšie nastavíme obmedzenia, tým vyšší stupeň ochrany dosiahneme.

2. Zraniteľnosť v ceste k súboru

Zraniteľnosť v ceste k súboru spočíva vo vybočení z predpokladaných stránok, ktoré mali byť zobrazené a miesto nich sú načítané systémové súbory, prípadne škodlivé súbory z iných lokalít na internete.

V našom príklade je stránka page.php určená ako zobrazovač obsahu ktorý mu posunul iný skript na webe. Kvôli zlej implementácii je možné podsunúť reťazec, ktorý vypíše obsah súboru so zoznamom používateľov systému, prípadne je možné vložiť stránku z iného zdroja, napríklad www.evilsite.com/evil.php. Takto je možné, aby sa vykonal cudzí kód na zraniteľnom systéme.

Po navštívení nasledovnej adresy:

```
127.0.0.1/dvwa/vulnerabilities/fi/?page=/../../../../../../../../../../../../../../../../etc/passwd
```

sme dostali odpoveď obsahujúcu zoznam používateľov a ich skupín v hornej časti stránky.

Priblížená časť vyzerá nasledovne:

```
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/bin
/games:/bin/sh man:x:6:12:man:/var/cache/man:/bin/sh lp:x:7:7:lp:
/spool/uucp:/bin/sh proxy:x:13:13:proxy:/bin:/bin/sh www-data:x:33
/list:/bin/sh irc:x:39:39:ircd:/var/run/ircd:/bin/sh gnats:x:41:41:Gnats
libuuid:x:100:101::/var/lib/libuuid:/bin/sh syslog:x:101:103::/home/s
/bin/false whoopsie:x:104:107::/nonexistent:/bin/false landscape:x
test:x:1000:1000:test,,,:/home/test:/bin/bash
```

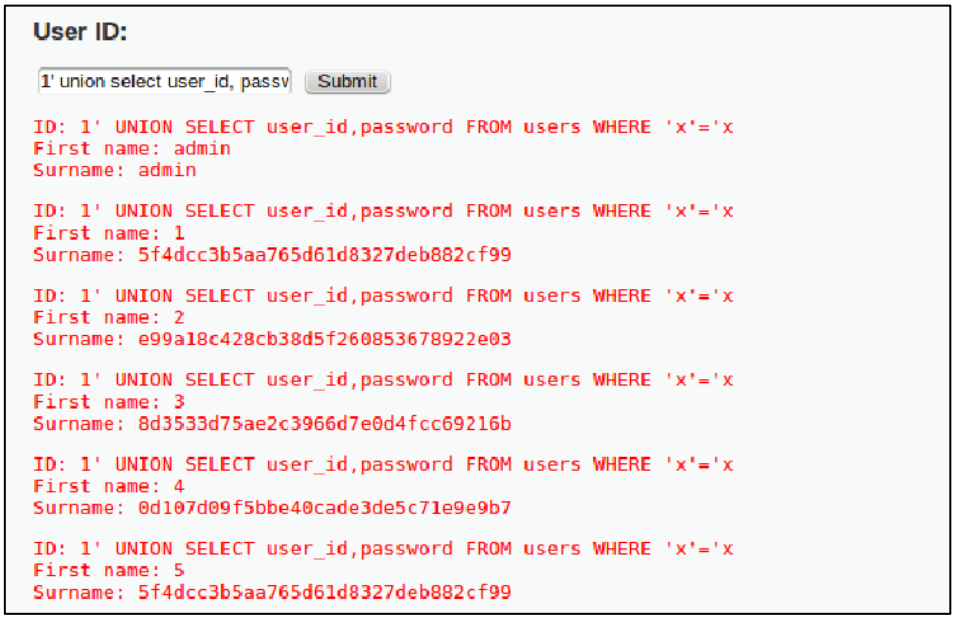
2.1. Ochrana

Ochrana pred týmto útokom spočíva v programátorskom ošetrení skriptu, ktorý vykonáva načítavanie iných stránok. Jedným z opatrení je obmedziť vstup akéhokoľvek iného reťazca znakov, ako sú povolené názvy súborov, ktoré má používateľ právo spustiť. Ďalšou ochranou je uzamknutie webserveru (Apache) v umelom koreňovom adresári (chroot).

3. SQL injection

SQL injection bol a stále je jeden z najpopulárnejších útočných vektorov, kde vďaka neošetrenému vstupu do databázy vie útočník vykonať SQL dotazy. Na základe výstupov z databázy dokáže útočník získať dáta zo systému (informácie o zákazníkoch, mená, heslá, čísla kreditných kariet, ...). V prípade niektorých konfigurácií databázového serveru útočník dokáže vykonávať systémové príkazy s oprávneniami SQL servera.

V našom príklade sme k požadovanému vstupu čísla používateľa, o ktorom chceme získať údaje, pridali aj SQL dopyt (1 ‘ union select user_id, password from users where ‘x‘ = ‘x), na základe ktorého sme vypísali obsah stĺpcov v tabuľke ku ktorým pôvodne nebolo možné pristupovať.



3.1. Ochrana

Ochrana pred SQL injection spočíva v zabránení vloženia SQL kódu do spracúvaného poľa aplikácie. Najsilnejšiu ochranu ponúkajú takzvané predpripravené výrazy (prepared statements), ktoré obmedzujú vstup od používateľa. Je možné rozlíšiť jednotlivé parametre ktoré vstupujú do SQL reťazca a obmedziť ich (integer, string...) tak, aby nebolo možné zadávať čokoľvek iné.