Command Injection

Command Injection je útok, při kterém je cílem vykonání neoprávněných příkazů na operačním systému. Útočník zneužívá nedostatečné zabezpečení aplikace, která nesprávně filtruje vstupy a dovoluje „přilepit“ systémové příkazy k těm legitimním.

Tato zranitelnost může vést k úplnému ovládnutí serveru, krádeži dat nebo smazání celého systému.

Jak útok probíhá?

Představte si webovou aplikaci, která umožňuje správci sítě otestovat dostupnost jiného serveru pomocí příkazu `ping`.

Původní kód na pozadí (PHP příklad):

```
$target = $_GET['ip'];
system("ping -c 4 " . $target);
```

Normální použití:

Uživatel zadá IP: `8.8.8.8` Systém vykoná: `ping -c 4 8.8.8.8`

Útok (Injection):

Útočník zadá do pole pro IP: `8.8.8.8 ; cat /etc/passwd` Systém vykoná: `ping -c 4 8.8.8.8 ; cat /etc/passwd`

Díky středníku (separátor příkazů v Linuxu) systém nejprve vykoná ping a hned poté vypíše obsah souboru s uživatelskými účty.

Klíčové techniky a operátory

Útočníci používají různé znaky v závislosti na operačním systému, aby řetězili příkazy:

Operátor	Funkce	Systém
;	Spustí druhý příkaz po prvním.	Linux / Unix
&	Spustí příkaz na pozadí.	Linux / Windows

Operátor	Funkce	Systém
&&	Spustí druhý příkaz pouze pokud první uspěje.	Linux / Windows
|	Pošle výstup prvního příkazu do druhého (Pipe).	Linux / Windows
||	Spustí druhý příkaz pouze pokud první selže.	Linux / Windows
` nebo **$( )**	Vykonání vnořeného příkazu.	Linux / Unix

Důsledky útoku

- **Exfiltrace dat:** Útočník může číst konfigurační soubory, databáze nebo zdrojové kódy.
- **Vzdálený přístup (Reverse Shell):** Útočník spustí příkaz, který server donutí připojit se k jeho počítači a předat mu ovládání (příkazovou řádku).
- **Lateral Movement:** Útočník využije napadený server jako odrazový můstek pro útok na další počítače ve vnitřní síti.
- **Destrukce:** Smazání databází nebo celého disku (např. pomocí příkazu `rm -rf /`).

Jak se bránit?

Obrana proti Command Injection stojí na třech pilířích:

```
1. **Vyhýbání se systémovým voláním:** Pokud je to možné, nepoužívejte
funkce jako ''system()'', ''exec()'' nebo ''passthru()''. Většina
programovacích jazyků má vestavěné funkce (API), které nepotřebují volat
shell (např. pro práci se soubory nebo sítí).
2. **Validace vstupu (White-listing):** Povolte pouze znaky, které
očekáváte. Pokud očekáváte IP adresu, povolte pouze čísla a tečky. Vše
ostatní zahoďte.
3. **Použití vestavěných funkcí pro únik (Escaping):** Pokud musíte volat
shell, použijte funkce, které nebezpečné znaky zneškodní (např.
''escapeshellarg()'' v PHP).
4. **Princip nejnižších privilegií:** Webový server by nikdy neměl běžet pod
účtem správce (root/admin). Pokud je napaden, útočník má omezené možnosti.
```

Související pojmy: SQL Injection, Malware, Rootkit, Shell, PHP, Kybernetická bezpečnost, Server.