Wytyczne dla uzyskania wysokiego poziomu bezpieczeństwa serwera aplikacyjnego

W zakresie bezpieczeństwa aplikacji webowych korzystamy z naszego kilkunastoletniego doświadczenia w tworzeniu aplikacji webowych (jako firma BetaSoft na rynku aplikacji webowych od 2003 r.), a kompetencje naszych pracowników potwierdzone są indywidualnymi certyfikatami uzyskiwanymi podczas szkoleń m.in. współpraca z firmą Niebezpiecznik.pl i konferencji (od początku powstania jesteśmy corocznym sponsorem PHPCon - największej konferencji poświęconej PHP).

Wdrożenia realizowane przez naszą firmę objęte są udokumentowanymi metodykami Agile i Prince, a pracownicy je wykonujący posiadają odpowiednie certyfikaty: Agile / Prince Foundation / Practitioner. Dzięki temu jesteśmy w stanie zachować najwyższe standardy bezpieczeństwa, nie tylko w odniesieniu do aplikacji i systemu operacyjnego, ale również w zakresie realizowanych usług. Potwierdzeniem naszych kompetencji są udane i referencyjne wdrożenia w wymagającym pod tym względem sektorze publicznym i obronnym.

W celu zapewnienia bezpieczeństwa i ciągłości działania systemu eDokumenty, w tym udostępnienie usługi w sieci publicznej, zalecane jest wprowadzenie poniższych zasad.

1. Bezpieczeństwo fizyczne (nie dotyczy wirtualnych maszyn)

- Dostęp do BIOS'u (Basic Input/Output System) serwera powinien być zabezpieczony hasłem.
- Możliwość uruchamiania systemu z urządzeń innych niż wewnętrzny dysk twardy (np. USB, CD-ROM, floppy drive) powinna być zablokowana.
- Bootloader systemu powinien być zabezpieczony hasłem (np. GRUB).

2. Bezpieczeństwo systemu operacyjnego Linux

- W systemie operacyjnym powinno być zainstalowane tylko niezbędne oprogramowanie wymagane do działania systemu.
- Niepotrzebne usługi powinny być wyłączone, a oprogramowanie odinstalowane (np. xserver, udostępnianie systemów plików NFS/SMB, narzędzia do kompilacji gcc/g++ itp.).
- Dla punktów montowań takich jak: /, /var, /var/log, /usr, /tmp, /home powinny być stworzone osobne systemy plików.
  Systemy plików powinny być montowane z odpowiednimi opcjami (np. dla /tmp nodev,nosuid,noexec).
- System operacyjny powinien być na bieżąco aktualizowany (szczególnie dotyczy to aktualizacji bezpieczeństwa).
- Dostęp do Cron'a dla nieuprzywilejowanych użytkowników powinien być ograniczony (/etc/cron.allow, cron.deny).
- Wiadomość powitalna wyświetlana przy próbie logowania powinna jednoznacznie informować, że dostęp od serwera jest dozwolony tylko dla autoryzowanych osób.

3. Polityka haseł

- Hasła użytkowników systemu powinny być silne tzn. nie powinno być krótsze niż 10 znaków, zawierać duże i małe litery oraz cyfry i znaki specjalne. Wymagania dotyczące haseł powinny być weryfikowane przez biblioteki takie jak libpam-cracklib i/lub libpam-pwquality.
- Hasła powinny być regularnie zmieniane (np. co 90 dni).

4. Zabezpieczenia sieci

4.1 Firewall

- Dostęp do serwera przez sieć powinien być zabezpieczony firewall'em, jeśli to możliwe serwer powinien znajdować się w DMZ
- Usługa SSH powinna być dostępna tylko dla ograniczonej liczby adresów IP lub wyłącznie dla połączeń VPN.
- Dostęp do HTTPS powinien być ograniczony dla wybranych prefiksów adresów IP (geolokalizacja adresów IP).
- Protokół IPv6 jeśli nie jest używany powinien być wyłączny.
- Przekazywanie pakietów między interfejsami sieciowymi powinno być wyłączone (IP forwarding).

4.2 Zdalny dostęp przez SSH

- Domyślny port 22 powinien być zmieniony na inny np. 2134.
- Logowanie jako użytkownik root powinno być zablokowane.
- Autentykacja powinna być dokonywana z użyciem pary kluczy (public/private).
- Usługa SSH powinna być zabezpieczona przed atakami typu brute-force (np. przez wykorzystanie aplikacji fail2ban).

5. Logowanie zdarzeń

- Informacje pochodzące ze standardowych źródeł powinny być rejestrowane za pomocą programu syslog. Zalecane jest logowanie na zdalny serwer logów.
- Logi powinny być na bieżącą przeglądane w celu wykrycia potencjalnych problemów (logwatch, swatch).
- Zegar systemowy powinien być prawidłowo ustawiony (zalecane jest używanie protokołu NTP).

6. Kopie bezpieczeństwa

- Kopie baz danych powinny być robione za pomocą dedykowanych narzędzi. W przypadku kopiowania danych serwera wymagane jest kopiowanie katalogu /home/edokumenty oraz kopii bazy znajdującej się w np. /home/edokumenty/backup_sql
- W przypadku serwerów wirtualnych najlepszym rozwiązaniem jest wykonywanie kopii całej maszyny np aplikacją Veem
- Kopie system operacyjnego powinny być robione na zdalne host lub zewnętrzne nośniki.

7. Dodatkowe zabezpieczenia

- System operacyjny powinien być zabezpieczony mechanizmami typu Mandatory Access Control (MAC) (np. AppArmor).
- Zalecane jest uszczelnienie słabych punktów systemu za pomocą narzędzi z pakietu Bastille Linux.
- Zaleca się dokonywanie testów integralności plików systemu za pomocą narzędzi takich jak AIDE, Tripwire.
- Komunikacja sieciowa serwera może być dodatkowo zabezpieczona za pomocą systemów IPS/IDS (np. Snort).

8. Zabezpieczenie oprogramowania serwera aplikacyjnego

Apache

- conf-enabled/security.conf

```
ServerSignature Off
Header set X-Content-Type-Options: "nosniff"
Header set X-Frame-Options: "sameorigin"
Header always set Referrer-Policy "same-origin"
```

9. Zabezpieczenia aplikacji eDokumenty/Ready_™

- Opcje do włączenia/sprawdzenia w config.inc
  - define('CRAZY_ABOUT_SECURITY_MODE', TRUE); - sprawdzanie sesji, ip
  - define('PARTIAL_PASSWORD_VISIBLE_CHARACTERS', TRUE); Dla trybu logowania poprzez losowe znaki z hasła. Stała określająca ile znaków z hasła musi podać użytkownik
  - define('CSP_HEADER', => 'obrazki.moja_domena.pl'?); Definiowanie dodatkowych opcji w nagłówku Content-Security-Policy, dodatkowe adresy odzielamy spacją

10. Weryfikacja checklisty OWASP

Kluczowe wytyczne standardu bezpieczeństwa OWASP zostały zebrane są w formie checklisty, wg której weryfikowane są tworzone komponenty systemu. Checklista w swoim zakresie w większości odpowiada obecnemu standardowi OWASP Level 2.

Przydatne linki

Zabezpieczanie serwera - wybrane skrypty