BO·TT·EGA
IT minds

Program szkolenia:

Security Aware Developer - Szkolenie z bezpieczeństwa aplikacji dla programistów

Informacje:

Nazwa:	**Security Aware Developer - Szkolenie z bezpieczeństwa aplikacji dla programistów**
Kod:	**sec-SAD**
Kategoria:	Bezpieczeństwo
Odbiorcy:	testerzy, developerzy, architekci
Czas trwania:	1 dzień
Forma:	50% wykładów, 50% dyskusji i ćwiczeń

Warsztaty z bezpieczeństwa dla zespołów projektowych - podnoszące świadomość problemów, operujące na realnych przykładach błędów oraz kupione na wdrożeniu właściwego poziomu zabezpieczeń.

Utrzymanie bezpieczeństwa aplikacji polegające jedynie na końcowej weryfikacji zabezpieczeń i usuwaniu defektów ciągnie za sobą wiele dodatkowych kosztów (konieczność ponownej przebudowy aplikacji, kary umowne, czas spędzony na usuwaniu błędów). Takie podejście może być również źródłem potencjalnych kłopotów jak np. wyciek wrażliwych danych.

Wyczulając zespół projektowy na kwestie bezpieczeństwa skracamy proces produkcji oprogramowania obniżając jego koszty.

Warsztaty Security Aware Developer zostały opracowane na podstawie setek rozmów z deweloperami jak i całymi zespołami projektowymi. Kładziemy nacisk nie tylko na unikalną formułę, ale też na wyjątkową praktyczność. Dlatego składowymi każdego scenariusz są:

- przedstawienie aplikacji - na której będzie bazował scenariusz,
- omówienie łańcucha podatności i jego realnych skutków,
- wyłonienie przez uczestników możliwych sposobów zabezpieczeń,
- analiza zaproponowanych zabezpieczeń z doświadczonym pentesterem.

Scenariusze przedstawione w trakcie warsztatów stworzone są na bazie naszych doświadczeń i opracowane w atrakcyjną graficznie formę zeszytu. Nie jest to kolejne szkolenie “do przeklikania” na komputerze. Grupa pracuje na specjalnie przygotowanych materiałach ułatwiających skupienie, bez użycia komputera.

Głównymi celami szkolenia są:

- Podnoszenie świadomości developerów, architektów, testerów odnośnie problemów bezpieczeństwa aplikacji,
- Pokazywanie w formie scenariuszy, realnych, z życia wziętych przykładów,
- Ustalenie najefektywniejszych rekomendacji poprawiających bezpieczeństwo danego rozwiązania.

Na podstawie informacji zwrotnej widzimy, że poruszane przykłady, dyskusje oraz praca uczestników związana z wymyślaniem poprawek i zabezpieczeń realnie wpływa na podniesienie świadomości wśród uczestników oraz powoduje, że po powrocie do codziennej pracy chcą realizować pomysły związane z bezpieczeństwem aplikacji, które wykiełkowały w trakcie szkolenia.

Zalety szkolenia:

- Szkolenie prowadzone jest przez byłych programistów, którzy znają wyzwania i problemy "od podszewki"
- Podczas szkolenia omawiane są wyłącznie przypadki "z życia wzięte"
- Formuła szkolenia - nie jest to szkolenie w formie nudnego wykładu. W trakcie szkolenia uczestnicy mają rzeczywisty wpływ na jego przebieg i rodzaj przekazywanych treści
- Skupiamy się na zapobieganiu podatnościom, a nie na nauce ich testowania
- W trakcie szkolenia jest czas na omówienie problemów z bezpieczeństwem aplikacji z którymi przychodzą uczestnicy

BOTTEGA IT minds

Szczegółowy program:

1. XXE

- 1.1. Błąd kontroli dostępu
- 1.2. Nieprawidłowe przetwarzanie plików XML
- 1.3. Spowodowanie niedostępności aplikacji jednym żądaniem HTTP
- 1.4. Wywołanie dowolnej komendy na serwerze
- 1.5. Wykradnięcie plików z serwera
- 1.6. Zapobieganie
  - 1.6.1. Konfiguracja parsera
  - 1.6.2. Hardening środowiska

2. XSS

- 2.1. Błąd walidacji danych wejściowych
- 2.2. Stored XSS w szablonach Angular
- 2.3. Trwała awaria aplikacji
- 2.4. Zapobieganie:
  - 2.4.1. Nieustająca walka z XSS’ami
  - 2.4.2. Znaj dane użytkownika swego
  - 2.4.3. Wiedz gdzie i w jakim kontekście są zwracane

3. PHISHING czyli jak przejąć kontrolę nad kodem

- 3.1. Analiza specyfiki pracy firmy deweloperskiej
- 3.2. Uzyskanie loginu i hasła do jednego z serwisów deweloperskich
- 3.3. Nadużycie SSO
- 3.4. Uzyskanie dostępu do kodu źródłowego
- 3.5. Zapobieganie:

BOTTEGA IT minds

3.5.1. Jak utrudnić przeprowadzanie ataków

3.5.2. Na co zwracać uwagę

4. ESKALACJA

4.1. Nadmiarowy moduł aplikacji

4.2. Domyślne (niebezpieczne) ustawienia

4.3. Zapisywanie wrażliwych danych do logów

4.4. Uzyskanie uprawnień administratora aplikacji

4.5. Zapobieganie:

4.5.1. Minimalizacja powierzchni ataku

4.5.2. Weryfikacja ustawień związanych z bezpieczeństwem

5. SQL INJECTION

5.1. Aplikacja legacy

5.2. Chałupnicza kryptografia

5.3. Skrót versus podpis

5.4. SQL Injection

5.5. Ominięcie zabezpieczeń i pobieranie danych

5.6. Zapobieganie:

5.6.1. Jak implementować kryptografię poprawnie

5.6.2. Wachlarz rozwiązań dla SQL Injection

6. BANK

6.1. Case study jednego z polskich banków

6.2. Znany defekt w komponencie

6.3. RCE - omówienie na przykładach

6.4. Przejęcie kontroli nad frontendem

6.5. Modyfikacja JavaScript

BOTTEGA IT minds

6.6. Obejście dodatkowej autoryzacji

6.7. Kradzież środków z kont

6.8. Jak wyprowadzać środki z przejętych kont

6.9. Zapobieganie:

6.9.1. Bezpieczeństwo zewnętrznych bibliotek

6.9.2. Monitoring środowiska i aplikacji

6.9.3. Bezpieczeństwo własnego kodu