ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 2011
Seria: ORGANIZACJA I ZARZĄDZANIE z. 57 Nr kol. 1847

Ireneusz J. JÓŹWIAK, Michał KRUPKA
Politechnika Wrocławska
Wydział Informatyki i Zarządzania

BEZPIECZEŃSTWO APLIKACJI TWORZONYCH W SYSTEMIE FENON

Streszczenie. Artykuł opisuje system FENON jako narzędzie wspomagające tworzenie aplikacji webowych. Programista PHP, używając systemu FENON, skupia się w większym stopniu na procesie przetwarzania informacji niż na pisaniu kodu. System tworzy odpowiednie środowisko webowe zgodne z przedstawionym modelem. Generowane przez system środowisko spełnia podstawowe wymogi bezpieczeństwa. Zgodnie z opublikowanymi przez OWASP (The Open Web Application Security Project) podpunktami, dotyczącymi błędów przy tworzeniu aplikacji webowych, w artykule omawiany jest problem błędów, które mogą wystąpić oraz jego rozwiązanie w systemie FENON.

PROTECTION OF APPLICATIONS CREATED IN FENON SYSTEM

Summary. This paper describes FENON system as a tool which is helpfully in web applications creating. In FENON system a programmer creates an information model without concentrate on a program code. Environment which is created by our system is fully protected against typical attacks and threats. In accordance with OWASP (The Open Web Application Security Project) report this paper describes problems and solution which is implemented in FENON.

1. Wprowadzenie

Aplikacje webowe stają się coraz bardziej popularne. Programy działające w „oknie przeglądarki" ewoluowały od prostych formularzy do zaawansowanych aplikacji [2]. Techniki i rozwiązania obecnie stosowane sprawiają, że programy obsługiwane przez przeglądarkę internetową nie różnią się funkcjonalnością od wszystkich obecnie znanych programów instalowanych lokalnie na komputerach. Tworzenie aplikacji w architekturze klient – serwer,

gdzie klientem jest dowolna nowoczesna przeglądarka internetowa, niesie ze sobą wiele korzyści. Najistotniejsze to:

1. **Przenośność programów**: użytkownik do korzystania z aplikacji używa głównie przeglądarki internetowej, programy funkcjonujące za jej pomocą działają w każdym systemie operacyjnym wyposażonym w przeglądarkę.
2. **Dostępność**: zapewniając serwerowi aplikacji webowej dostęp do sieci WWW, możemy umożliwić korzystanie z aplikacji z dowolnego miejsca.
3. **Niskie wymagania sprzętowe po stronie użytkownika**: komputer użytkownika musi spełniać jedynie minimalne wymagania zainstalowanej przeglądarki internetowej.
4. **Darmowe narzędzia**: istnieje wiele darmowych narzędzi, wtyczek, bibliotek, a nawet systemów wspomagających proces tworzenia aplikacji webowej [3].

Z aplikacji internetowych korzystamy kupując książki, płacąc rachunki, sprawdzając otrzymane wiadomości. Mnogość zalet rozwiązań webowych, przy jednoczesnym relatywnym niskim koszcie tworzenia i wdrożenia aplikacji, sprawia, iż zastępują one coraz więcej dotychczasowych rozwiązań (webowy edytor tekstu, arkusz kalkulacyjny). Proponowane już jest zastąpienie całego systemu operacyjnego jedynie przeglądarką internetową (http://dev.chromium.org/chromium-os).

Wychodząc naprzeciw rosnącej popularności aplikacji internetowych, zaimplementowany został system FENON, a jego zadaniem jest wspomaganie programisty PHP podczas tworzenia aplikacji.

Niniejszy artykuł skupia się na przedstawieniu systemu FENON jako bezpiecznego narzędzia ułatwiającego implementację aplikacji PHP. W artykule przedstawione zostały podstawowe zagadnienia związane z systemem, jego model, architektura oraz użyte techniki i zabezpieczenia, zwiększające poziom bezpieczeństwa generowanych aplikacji. Praca oparta jest na publikacji OWASP (*Open Web Application Security Project*), w której przedstawione zostały podstawowe błędy popełnianie przez programistów podczas implementacji aplikacji webowej.

System FENON został zaimplementowany w taki sposób, aby generowane przez niego aplikacje nie miały wskazanych błędów, obniżających znacznie poziom bezpieczeństwa.

2. System FENON – ujednolicony model systemu informacyjnego i model aplikacji

System FENON ma model porządkowujący wszystkie przechowywane w aplikacji dane, a jego podstawową jednostką organizacji są dokument i lista dokumentów.

W skład dokumentu wchodzi wiele unikalnych pól, które przechowują wszystkie podstawowe informacje, np. tytuł, opis, datę itp. Dodatkowo może być z nim powiązana lista załączników, będąca zbiorem innych dokumentów.

Lista to uporządkowany zbiór dokumentów, mogący występować jako osobny element albo jako część innego dokumentu. Taka metoda organizowania informacji pozwala na przechowywanie dowolnych danych i przetwarzanie ich w jednolity sposób.

Wobec każdego dokumentu możemy wykonać jedną z czterech czynności:

a) wyświetlenie wybranego dokumentu,
b) edycję wybranego dokumentu, w tym usunięcie,
c) utworzenie nowego dokumentu,
d) wyświetlenie listy dokumentów.

W systemach magazynujących informacje zasadniczą funkcją jest ich skuteczne wyszukiwanie. Składa się ono z dwóch etapów:

a) opisania poszukiwanego dokumentu i uściślenia zapytania,
b) wyświetlenia listy wyników wyszukiwania.

W systemie FENON dostępne są dwie metody wyszukiwania:

1. Zawężanie listy przez zdefiniowanie określonych kryteriów, np.: określenie przedziału czasowego utworzenia dokumentu, nazwania części dokumentu. Funkcja ta dostępna jest bezpośrednio z poziomu listy dokumentów.
2. Formularz, w którym użytkownik ma więcej opcji zdefiniowania kryteriów wyszukiwania. Otrzymane dane są przedstawiane w postaci podstawowej listy systemu FENON.

Ważną częścią systemów informacyjnych jest możliwość tworzenia raportów. W systemie FENON raport złożony jest z dwóch podstawowych części. Są to:

a) lista dokumentów spełniających określone kryteria,
b) dokument zawierający i podsumowujący np.: sumę, średnią, odchylenie standardowe oraz inne pola wyliczane na podstawie dokumentów z listy.

Systemy informacyjne gromadzą dane dotyczące działalności całej organizacji. Nie wszyscy użytkownicy systemu powinni mieć dostęp do całości zasobów, dlatego wprowadza się osobne konta oraz grupy dostępów.

W systemie FENON mamy możliwość utworzenia 64 grup zarządzających dostępem do informacji. Użytkownicy oraz dokumenty mogą zostać przypisane do wielu grup. Każdy dokument zawiera informację o możliwościach danej grupy – czy ma ona prawo tylko do odczytu czy też do odczytu i zapisu. Dokument przechowuje także informację o właścicielu, mającym zawsze prawa do odczytu i zapisu.

Aplikacje tworzone za pomocą systemu FENON oparte zostały na trójelementowym modelu, który tworzą następujące elementy:

1. **Widok** – jest zestawem danych (zbiorem pól), tworzących rozumiany ogólnie Dokument. Widokiem mogą być np. dane personalne osoby czy też notatka w kalendarzu, zawierająca dane o planowanym spotkaniu itp. Widok może wiązać wiele tabel z bazy danych, tak aby tworzyły logiczną i czytelną całość dla użytkownika. Za pomocą Widoku komunikujemy się z Bazą Danych i powinien to być jedyny sposób komunikacji. W Widoku składowane są również dane wpisane przez użytkownika, pochodzące z jego interakcji z systemem. Są to dane pochodzące z przesyłanych formularzy.
2. **Kompozytor** – tworzy dokumenty przedstawiane użytkownikowi. Służy programiście do tworzenia dokumentów, które wykorzystuje użytkownik, oraz do interfejsów, z których użytkownik korzysta, i potrafi zapewnić mu całą obsługę dokumentu. Kompozytor potrafi, w niewymagających innowacji warunkach, opracować cały dokument bez ingerencji ze strony programisty. Do standardowych odmian dokumentów zaliczymy: Listę Elementów, Podgląd Dokumentu, Edycję Dokumentu, Formularz Utworzenia Nowego Dokumentu, Zapisanie Dokumentu (edytowanego bądź też nowego), Usunięcie Dokumentu. Kiedy programista chce utworzyć dokument odznaczający się nietypowymi cechami, wtedy Kompozytor udostępnia metody wspomagające jego pracę.
3. **Kontroler** – tworzy tzw. logikę biznesową systemu, by wszystkie wykonywane czynności i procesy miały swoją podstawę w rzeczywistości. Dane przetwarzane przez Kontrolera mogą pochodzić z Widoku lub mogą zostać wprowadzone przez użytkownika. Kontroler może nimi operować. Dane przechodzące przez Kontroler mogą przepływać w dwóch kierunkach: z Widoku do Kompozytora (czyli do użytkownika) lub z Kompozytora do Widoku (czyli od użytkownika).

3. Kontrola danych wejściowych

Aplikacja webowa, w odróżnieniu od statycznej strony internetowej, zapewnia interakcję z użytkownikiem. Wybierając odpowiednie elementy menu i wprowadzając dane, użytkownik wpływa na treść prezentowanej strony. Pełne zaufanie do poprawności informacji przesyłanej przez użytkownika może nieść za sobą bardzo poważne konsekwencje.

Według raportu OWASP brak kontroli danych wejściowych jest najczęstszą przyczyną zmniejszenia poziomu bezpieczeństwa aplikacji webowej [1]. Tworząc system informacyjny,

należy walidować wszystkie przesyłane przez użytkownika dane oraz dbać o ich integralność i zgodność z zamodelowaną logiką biznesową.

W tworzonych aplikacjach dane przesyłane przez użytkownika do systemu powinny być dokładnie sprawdzane. Zaleca się ich badanie pod względem zgodności ze wzorcem (typ, długość, zgodność z określonym wyrażeniem regularnym), sumą kontrolną, np. sumy kontrolne dla atrybutów, takich jak NIP, PESEL, numer konta. Do podstawowych strategii postępowania z danymi przysyłanymi przez użytkownika należą [1]:

a) **Strategia białej listy** – istnieje zdefiniowana lista akceptowanych danych. Wszystkie przesyłane dane nieznajdujące się na liście są odrzucane.
b) **Strategia czarnej listy** – istnieje zdefiniowana lista nieakceptowanych danych. Tylko dane nieznajdujące się na liście są akceptowane i przesyłane do dalszej analizy.
c) **Edycja informacji (z listami białą i czarną)** – bardzo skuteczna metoda zapewniająca wysoki poziom bezpieczeństwa oraz poprawność funkcjonowania systemu.

Dane przesyłane przez użytkownika poddawane są wstępnej edycji, np. usunięciu znacznika *<script>* z wypowiedzi, zamianie „„ na „„.

W systemie FENON zostało zaimplementowanych wiele metod kontrolujących przesyłane dane, które zapewniają nie tylko wyższy poziom bezpieczeństwa tworzonych aplikacji, lecz także ułatwiają korzystanie z niej. Do podstawowych metod, możliwych do wybrania podczas modelowania aplikacji, należą:

a) **Automatyczna walidacja**. Przypisanie pola formularza (*input*) do odpowiedniej klasy (np. *date*, PESEL, NIP, *number, reg*) powoduje automatyczne wygenerowanie na stronie kodu walidującego formularz (zarówno po stronie użytkownika, jak i serwera).
b) **Ułatwienie wprowadzania daty**. Dla pól typu *date* istnieje możliwość automatycznego wyświetlenia kalendarza podczas edycji. Ułatwia to wprowadzenie daty w odpowiednim formacie.
c) **Dodawanie słowników**. System FENON pozwala tworzyć zbiory wyrażeń możliwych do wyboru podczas edycji danego elementu. Elementy słownika wyświetlane są na stronie w postaci rozwijanej listy wyboru, z podpowiedzią lub bez niej.
d) **Edycja przesyłanej informacji**. System FENON domyślnie usuwa z przesyłanych pól formularza znaki specjalne PHP – »\« (odwrócony ukośnik), »„"« (cudzysłów), »'« (apostrof). Domyślnie dla pól typu liczbowego w formularzach »,« zamieniany jest na ».«.
e) **Bezpieczny adres URI**. Adres internetowy aplikacji może potencjalnemu atakującemu ułatwić niepożądaną manipulację danymi. URI systemu FENON zawiera tylko niezbędne pola służące do nawigacji. Za pomocą adresu (tablicy GET) przekazywana jest tylko informacja o żądaniu wyświetlenia danego widoku. Pozostałe dane, nie-

zbędne do poprawnego funkcjonowania systemu, przesyłane są w bardziej dyskretny sposób (tablica POST).

4. Ataki typu XSS

Atak XSS (Cross Site Scripting) oparty jest na przekazaniu kodu (zazwyczaj JavaScript) do aplikacji w nieprzeznaczonym do tego miejscu. Aplikacja niekontrolująca danych wprowadzanych do systemu potraktuje kod jak zwykły tekst. Dalsza praca z zapisanym kodem może uniemożliwić korzystanie z aplikacji. Kod *JavaScript* zostanie wysłany do przeglądarki użytkownika (jak zwykły tekst) i wykonany zgodnie z zapisanymi instrukcjami. Skutkiem wykonania niepożądanego kodu mogą być przekierowanie na inną stronę, zawieszenie przeglądarki itp.

Atak XSS składa się z trzech faz:

1. wysłanie złośliwego kodu w możliwe do tego miejsce,
2. pobranie kodu przez aplikację i wysłanie go do przeglądarki użytkownika,
3. inne działanie związane z wykonaniem niepożądanego kodu.

System FENON zapewnia ochronę przed atakami typu XSS w sposób automatyczny.

Podczas implementacji systemu kładziony jest szczególny nacisk na kontrolę danych przesyłanych przez użytkownika. Stosując opisane w rozdziale 2 metody zarządzania informacją, niepożądany kod JavaScript zostałby zapisany jako zwykły tekst (znaczniki <script></script> zostałyby usunięte).

Aplikacje tworzone w systemie FENON kontrolują poprawność danych zarówno po stronie użytkownika, jak i serwera aplikacji. Użytkownik nie będzie miał aktywnego przycisku, wysyłającego formularz, do czasu poprawnego uzupełnienia danych (wykonania modyfikacji danych).

Po wysłaniu elementów formularza do serwera następuje ponowne sprawdzenie danych. Dopiero po dwukrotnym sprawdzeniu i ewentualnej modyfikacji danych wpływają one na zachowanie aplikacji.

Do kontroli większych partii tekstu (dokumentów, komentarzy) w systemie FENON został wykorzystany edytor WYSIWYG (*What You See Is What You Get*) FCKeditor (http://www.fckeditor.net/). Zapewnia on pełną kontrolę wprowadzanych danych oraz przyjazny i intuicyjny interfejs.

5. Wady wstrzyknięcia kodu (SQL, programu), przechowywanie danych

System FENON został napisany w języku PHP. Kod aplikacji tworzonych za jego pomocą również ma składnię tego języka. Do poprawnego funkcjonowania tych aplikacji niezbędna jest relacyjna baza danych, której wykorzystanie wpływa na wysoką wydajność tworzonych systemów. Domyślnie, wspieraną bazą danych jest baza MySQL [4].

Połączenie PHP i MySQL jest niewątpliwie najpopularniejszym tandemem do tworzenia aplikacji webowych [2]. Niepoprawna obsługa baz danych przez źle zaprojektowane i zaimplementowane aplikacje, powodując znaczne obniżenie poziomu bezpieczeństwa.

Funkcje odpowiedzialne za generowanie zapytań do baz danych lub wykorzystujące funkcje systemowe powinny być napisane bardzo rozważnie. Nieodpowiednie filtrowanie parametrów przyjmowanych przez te funkcje daje potencjalnemu crackerowi możliwość tzw. wstrzyknięcia kodu, dzięki któremu może on uzyskać dostęp do wszystkich tabel bazy danych. System FENON został zaprojektowany z myślą o zapewnieniu jak największego poziomu bezpieczeństwa w aplikacjach działających pod jego kontrolą. Zabezpieczanie bazy danych przed niechcianym odczytem informacji jest realizowane na dwóch poziomach.

Pierwszym poziomem zabezpieczeń przed wstrzyknięciem kodu jest dokładna walidacja danych przesyłanych przez użytkownika. Zmienne wchodzące w skład zapytań do bazy danych są sprawdzane pod kątem zgodności ze wzorcem.

Drugim poziomem zabezpieczającym przed niepożądanym dostępem do bazy danych jest możliwość tworzenia wielu użytkowników bazy danych. Modelowanie praw dostępu w systemie jest pośrednio powiązane z modelowaniem uprawnień w bazie danych. Użytkownik o niskim poziomie uprawnień nie dokona przeglądu zupełnego bazy danych. Zarówno konto systemu FENON, jak i bazy danych nie pozwolą mu wyświetlić informacji, do których nie będzie miał uprawnień.

Aby utworzona aplikacja webowa była dostępna z dowolnego miejsca, musi znajdować się na serwerze z dostępem do sieci WWW. Podczas umieszczania gotowej aplikacji na serwerze należy zwrócić szczególną uwagę na jego poziom zabezpieczeń i konfigurację. Najlepsze techniki programistyczne nie będą w stanie zabezpieczyć naszych informacji, kiedy serwer obsługujący aplikacje będzie źle chroniony. Jeśli aplikacja przechowuje dane, których ujawnienie może nieść ze sobą bardzo duże konsekwencje, należy rozważyć dodatkowe opcje zabezpieczeń. Wykorzystanie protokołu SSL do nawiązywania bezpiecznych połączeń z aplikacją, skompilowanie kodu utworzonej aplikacji, przechowywanie informacji w zakodowanej formie są przykładowymi zabiegami zwiększającymi poziom bezpieczeństwa aplikacji internetowej.

6. Podsumowanie

Aplikacje webowe cieszą się coraz większą popularnością. Istnieje wiele narzędzi wspomagających ich tworzenie. FENON jest kompleksowym systemem umożliwiającym, w szczególnych przypadkach, wygenerowanie całego kodu aplikacji. Jest narzędziem odciążającym programistę podczas tworzenia programu. Korzystając z funkcji zawartych w systemie, programista nie tylko szybciej tworzy aplikację, ale tworzy również aplikację bezpieczną, odporną na większość popularnych ataków.

Projekt FENON jest projektem rozwojowym. Pracę nad nowymi modułami nieustannie trwają.

BIBLIOGRAFIA

1. The Open Web Application Security Project. Development Guide.
2. Luke W., Laura T.: PHP i MySQL. Tworzenie stron WWW. Helion, Gliwice 2005.
3. Strona główna PHP: http://php.net.
4. Strona główna MySQL: http://www.mysql.com.

Recenzent: Dr hab. inż. Jan Kałuski, prof. nzw. w Politechnice Śląskiej

Abstract

Web applications are more and more frequently used in different parts of economic and commercial activities. With their help we buy books, pay bills. This paper describes a FENON system which is helpfully in creating a web applications. FENON based on information model which consist a Document and Documents list. Our system implements a view, controller and creator (changed version of MVC). This paper describes the problem with controlling data sent by user and attacks following lack of such control (XSS and Injection attack). In detail are presented methods of data verification which are implemented in FENON.