Uniwersytet Ekonomiczny w Katowicach

Studia Ekonomiczne. Zeszyty Naukowe
Uniwersytetu Ekonomicznego w Katowicach
ISSN 2083-8611 Nr 390 · 2019

Informatyka i Ekonometria 15

Anna Sołtysik-Piorunkiewicz

Uniwersytet Ekonomiczny w Katowicach
Kolegium Informatyki i Komunikacji
Katedra Informatyki
anna.soltysik-piorunkiewicz@ue.katowice.pl

Monika Krysiak

Uniwersytet Ekonomiczny w Katowicach
Kolegium Informatyki i Komunikacji
Katedra Badań Operacyjnych
monika.krysiak@edu.uekat.pl

INŻYNIERIA ZABEZPIECZEŃ APLIKACJI INTERNETOWYCH NA PODSTAWIE ANALIZY ZAGROŻEŃ I REKOMENDACJI OWASP

Streszczenie: W artykule przedstawiono zagrożenia bezpieczeństwa aplikacji internetowych w projektowaniu i budowie systemów informatycznych w oparciu o wytyczne wynikające z inżynierii bezpieczeństwa oprogramowania. Zidentyfikowano różnorodność i zmienność zagrożeń zabezpieczeń aplikacji internetowych. Celem zaprezentowanych badań jest analiza trendu występowania zagrożeń aplikacji internetowych na podstawie danych fundacji OWASP opublikowanych w latach 2003-2017. W pierwszym punkcie artykułu przedstawiono rolę i zadania fundacji OWASP na tle wytycznych opracowanych przez organizacje zajmujące się bezpieczeństwem aplikacji internetowych. W drugim scharakteryzowano najczęstsze zagrożenia bezpieczeństwa aplikacji internetowych. W trzeciej części dokonano analizy występowania i porównano częstość zagrożeń aplikacji internetowych w latach 2003-2017, a w czwartej przedstawiono sposoby zabezpieczenia aplikacji internetowych oraz rekomendacje do uwzględnienia w projektowaniu i budowie systemów informatycznych z zastosowaniem aplikacji internetowych oraz podczas ich eksploatacji.

Słowa kluczowe: inżynieria zabezpieczeń, bezpieczeństwo aplikacji internetowych, zagrożenia internetowe, bezpieczeństwo danych i informacji, OWASP.

JEL Classification: L86.

Wprowadzenie

Inżynieria zabezpieczeń dotyczy ochrony danych i informacji w systemach informatycznych z zastosowaniem różnych metod i technik, których źródłem są wytyczne wynikające z norm ISO (International Organization for Standardization – Międzynarodowa Organizacja Normalizacyjna) i standardów oraz dobrych

praktyk zarządzania bezpieczeństwem w systemach informatycznych, np. ITIL (Information Technology Infrastructure Library – zbiór zaleceń, jak efektywnie i skutecznie oferować usługi informatyczne), COBIT (Control Objectives for Information and related Technology – standard, zbiór dobrych praktyk z zakresu IT Governance, które mogą być wykorzystywane głównie przez audytorów systemów IT).

Inżynieria bezpieczeństwa aplikacji internetowych obejmuje również rekomendacje opracowane przez organizację OWASP. Open Web Application Security Project (OWASP) jest przykładem instytucji zrzeszającej osoby prywatne, korporacje i organizacje, które wspierają swoim doświadczeniem aktualny stan wiedzy na temat bezpieczeństwa IT dzięki analizie przypadków, głównie w obszarze aplikacji internetowych. Dzięki zgromadzonej wiedzy opartej na analizie studiów przypadków powstało wiele metod, narzędzi, opisów w formie dokumentacji i artykułów, mówiących o tym, jak zarządzać bezpieczeństwem aplikacji w systemach informatycznych. W latach 2003-2017 na podstawie badań organizacji OWASP powstało siedem rankingów najczęstszych zagrożeń aplikacji internetowych. Każdy nowy obszar został zbadany i opisany tak, aby twórcy lub użytkownicy systemów informatycznych łatwiej i szybciej byli w stanie wykryć zagrożenia i na podstawie praktycznych zaleceń zabezpieczyć się przed nimi.

1. Charakterystyka OWASP

Open Web Application Security Project (OWASP) jest organizacją non profit otwartą dla osób interesujących się zabezpieczeniami oprogramowania, którzy równocześnie są zaangażowani z realizacje zadań związanych z inżynierią zabezpieczeń w praktyce. Została utworzona 9 września 2001 r., a w czerwcu 2011 r. została zarejestrowana w Europie. Ideą stowarzyszenia jest poprawa bezpieczeństwa przede wszystkim w obszarze aplikacji internetowych. Celem organizacji OWASP jest udostępnianie obiektywnych informacji na temat praktycznych zasad używania aplikacji internetowych dla zapewnienia bezpieczeństwa systemów informatycznych w szerokim kontekście do zarządzaniu systemami bezpieczeństwa organizacji. Dodatkowo badania prowadzone przez OWASP dotyczą wykorzystywania konkretnych technologii internetowych, jednakże OWASP nie wspiera konkretnych produktów i usług, starając się zagwarantować otwartość podczas prowadzenia badań i publikacji wyników [*About The Open Web Application Security Project*, b.r.].

Organizacja OWASP posiada ok. 100 lokalnych oddziałów na świecie. Każdy z lokalnych oddziałów jest prowadzony niezależnie i stara się samodzielnie publikować swoje materiały. Dodatkowo OWASP współpracuje również z innymi instytucjami zajmującymi się bezpieczeństwem, gdzie jednym z przykładów jest ISACA (Information Systems Audit and Control Association – międzynarodowe stowarzyszenie osób zajmujących się zawodowo zagadnieniami dotyczącymi audytem, konrolą oraz innymi aspektami zarządzania systemami informatycznymi) [Sejdak, 2013; *About Us*, b.r.].

2. Przegląd zagrożeń bezpieczeństwa aplikacji internetowych

Opierając się na najstarszych zestawieniach zagrożeń bezpieczeństwa aplikacji, dokonano przeglądu danych o odnotowanych błędach aplikacji internetowych udostępnionych przez OWASP w 2003 r. Do najważniejszych zagrożeń aplikacji internetowych zaliczono wówczas następujące czynniki: niepoprawne parametry (Invalid Parameters), nieprawidłowa kontrola dostępu (Broken Access Control), niepoprawna obsługa uwierzytelnienia i sesji (Broken Autenticathion and Session Management), skrypty międzyserwisowe (Cross Site Scripting <XSS, CSS>), przepełnienie bufora (Buffer Overflow), błędy „wstrzykiwania" (Injecion), niepoprawna obsługa błędów (Error Handling Problems), niepoprawne korzystanie z kryptografii (Insecure Use of Cryptography), błędy w administracji zdalnej (Remote Administration Flaws) oraz błędy konfiguracji serwera WWW i aplikacji Web and Application Server Misconfiguration). Następnie rok później powstało uaktualnienie, które składało się z sześciu zagrożeń zidentyfikowanych w 2003 r., natomiast jako nowe zidentyfikowano: niewłaściwe postępowanie z błędami (Impropre Error Handling), niezabezpieczone przechowywanie (Insecure Storage), odmowę usługi aplikacji (Application Denial of Service) i niezabezpieczone zarządzanie konfiguracją (Insecure Configuration Management) [*OWASP Top Ten*, b.r.].

W 2007 r. zidentyfikowano w ramach działań OWASP siedem nowych zagrożeń takich jak: wykonanie złośliwego pliku (Malicious File Execution), niezabezpieczone bezpośrednie odniesienie do obiektu (Insecure Direct Object Reference), skrypty fałszujące dostęp do witryny (Cross Site Request Forgery <CSRF, XSRF>), wyciek informacji i niewłaściwe postępowanie z błędami (Information Leakage and Improper Error Handling), niezabezpieczone przechowywanie danych kryptograficznych (Insecure Cryptographic Storage), niezabezpieczoną komunikację (Insecure Communications) i brak dostępu do ogra-

niczonego adresu URL (Failure to Restric URL Access) [*OWASP 2007 Top 10 Presentation*, 2009].

Trzy lata później, w 2010 r. wzięto dodatkowo pod uwagę: błędy w konfiguracji zabezpieczeń (Security Misconfiguration), niezabezpieczone referencje obiektów (Insecure Direct Object References) oraz niewłaściwe przekierowanie (Unvalidated Redirects and Forwards). Zagrożenie dziewiąte (niezabezpieczona komunikacja) zmieniono na niewystarczającą ochronę warstwy transportowej (Insufficient Transport Layer Protection) [Wichers, b.r.].

W 2013 r. uwzględniono w rankingu OWASP kolejne nowe zagrożenia, tj. ujawnienie danych wrażliwych (Sensitive Data Exposure), brak kontroli dostępu na poziomie funkcji (Missing Function Level Access Control) i używanie komponentów ze znanymi lukami w zabezpieczeniach (Using Components with Known Vulnerabilities) [*OWASP Top 10 – 2013*…, b.r.].

Różnorodność zagrożeń została potwierdzona w zestawieniach raportów opublikowanych przez OWASP w 2017 r. Pierwsza edycja raportu z 2017 r. objęła dodatkowo niewystarczającą ochronę przed atakami (Insufficient Attack Protection) i niechronione interfejsy API (Underprotected APIs) [*Top 10-2017 Top 10*, b.r.].

W drugiej edycji raportu z listopada 2017 r. można zauważyć kilka kolejnych zmian, takich jak: zewnętrzne pliki XML (XML External Entities), niezabezpieczona deserializacja (Insecure Deserialization) oraz niewystarczające rejestrowanie i monitorowanie (Insufficient Logging & Monitoring) [*OWASP Top 10 2017*…, b.r.].

3. Analiza czynników wpływających na bezpieczeństwo aplikacji

Obszar badań analizy czynników wpływających na bezpieczeństwo aplikacji internetowych obejmuje raporty OWASP opublikowane w latach 2003-2017, tzw. TOP 10 w kolejnych edycjach. Raporty te koncentrują się na atakach najczęściej odnotowanych i opisanych przez ekspertów zajmujących się bezpieczeństwem IT.

Na podstawie zestawień zagrożeń opublikowanych przez OWASP na przełomie minionych 15 lat dokonano analizy występowania różnorodnych czynników związanych z opisanymi formami ataków cybernetycznych. Zestawienie w tabeli 1 obejmuje główne czynniki zagrożenia związane zarówno z zarządzaniem danymi, jak i obsługą aplikacji internetowych.

Tabela 1. Przegląd czynników zagrożenia bezpieczeństwa w aplikacja internetowych

Lp.	Czynnik zagrożenia bezpieczeństwa aplikacji internetowych
1.	Niepoprawne parametry / dane wejściowe
2.	Nieprawidłowa kontrola dostępu
3.	Niepoprawna obsługa uwierzytelnienia i sesji
4.	Skrypty międzyserwisowe (XSS)
5.	Przepełnienie bufora
6.	„Wstrzyknięcia”
7.	Obsługa błędów
8.	Niepoprawne używanie kryptografii
9.	Błędy w administracji zdalnej
10.	Błędy konfiguracji serwera WWW i aplikacji
11.	Niewłaściwe postępowanie z błędami
12.	Niezabezpieczone przechowywanie
13.	Odmowa usługi aplikacji
14.	Niezabezpieczone zarządzanie konfiguracją
15.	Wykonanie złośliwego pliku
16.	Niezabezpieczone bezpośrednie odniesienie do obiektu
17.	Skrypty fałszujące dostęp do witryny (XSRF)
18.	Wyciek informacji i niewłaściwe postępowanie z błędami
19.	Niezabezpieczone przechowywanie danych kryptograficznych
20.	Niezabezpieczona komunikacja / niewystarczająca ochrona warstwy transportowej
21.	Brak dostępu do ograniczonego adresu URL
22.	Błędy w konfiguracji zabezpieczeń
23.	Niewłaściwe przekierowanie
24.	Ujawnienie danych wrażliwych
25.	Brak kontroli dostępu na poziomie funkcji
26.	Używanie komponentów ze znanymi lukami w zabezpieczeniach
27.	Niewystarczająca ochrona przed atakami
28.	Niechronione interfejsy API
29.	Zewnętrzne pliki XML
30.	Niezabezpieczona deserializacja
31.	Niewystarczające rejestrowanie i monitorowanie

Źródło: Opracowanie własne.

Na podstawie analizy wszystkich zagrożeń, jakie przedstawiono na listach TOP 10 według fundacji OWASP w latach 2003, 2004, 2007, 2010, 2013, Realise Case 1 (RC1) 2017 i Realise Case 2 (RC2) 2017, dokonano analizy najczęściej występujących zagrożeń cybernetycznych. Można zauważyć, iż niepoprawna obsługa uwierzytelnienia i sesji, skrypty między serwisowe i tzw. wstrzyknięcia są cały czas jednymi z najważniejszych zagrożeń. Kolejnymi częstymi zagrożeniami są: błędy w konfiguracji zabezpieczeń i używanie komponentów ze znanymi lukami w zabezpieczeniach. Dodatkowo zagrożenie związane z ujawnianiem danych wrażliwych zostało dodane do zestawienia zagrożeń dopiero w 2013 r. i w edycji drugiej (RC2) w 2017 r.

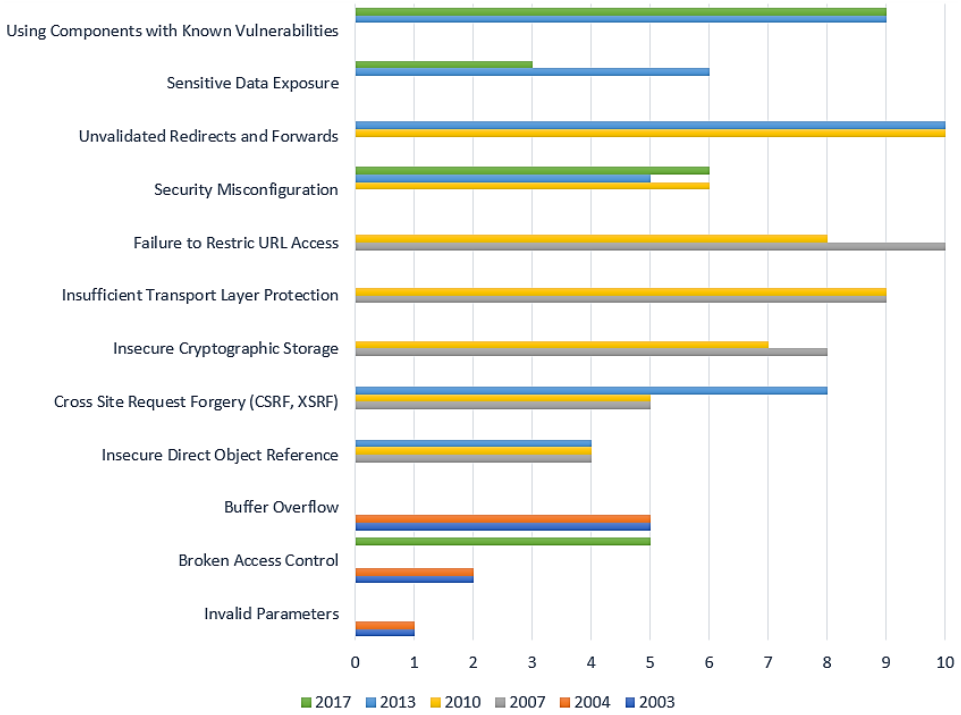
Różnorodność i zmienność występowania czynników związanych z zagrożeniami bezpieczeństwa aplikacji internetowej wpływa na stosowanie technologii internetowych do tworzenia aplikacji i metod zabezpieczenia – tabela 2. Pozycje w rankingu oznaczono symbolami od A1 do A10.

Tabela 2. Zmienność zagrożeń bezpieczeństwa aplikacji internetowych występujących w latach 2003, 2004, 2007, 2010, 2013, 2017

Nazwa zagrożenia	2003	2004	2007	2010	2013	2017 RC1	2017 RC2
Niepoprawne parametry / dane wejściowe	A1	A1					
Nieprawidłowa kontrola dostępu	A2	A2				A4	A5
Niepoprawna obsługa uwierzytelnienia i sesji	A3	A3	A7	A3	A2	A2	A2
Skrypty międzyserwisowe (XSS)	A4	A4	A1	A2	A3	A3	A7
Przepełnienie bufora	A5	A5					
„Wstrzyknięcia”	A6	A6	A2	A1	A1	A1	A1
Obsługa błędów	A7						
Niepoprawne używanie kryptografii	A8						
Błędy w administracji zdalnej	A9						
Błędy konfiguracji serwera WWW i aplikacji	A10						
Niewłaściwe postępowanie z błędami		A7					
Niezabezpieczone przechowywanie		A8					
Odmowa usługi aplikacji		A9					
Niezabezpieczone zarządzanie konfiguracją		A10					
Wykonanie złośliwego pliku			A3				
Niezabezpieczone bezpośrednie odniesienie do obiektu			A4	A4	A4		
Skrypty fałszujące dostęp do witryny (XSRF)			A5	A5	A8	A8	
Wyciek informacji i niewłaściwe postępowanie z błędami			A6				
Niezabezpieczone przechowywanie danych kryptograficznych			A8	A7			
Niezabezpieczona komunikacja / niewystarczająca ochrona warstwy transportowej			A9	A9			
Brak dostępu do ograniczonego adresu URL			A10	A8			
Błędy w konfiguracji zabezpieczeń				A6	A5	A5	A6
Niewłaściwe przekierowanie				A10	A10		
Ujawnienie danych wrażliwych					A6	A6	A3
Brak kontroli dostępu na poziomie funkcji					A7		
Używanie komponentów ze znanymi lukami w zabezpieczeniach					A8	A9	A9
Niewystarczająca ochrona przed atakami						A7	
Niechronione interfejsy API						A10	
Zewnętrzne pliki XML							A4
Niezabezpieczona deserializacja							A8
Niewystarczające rejestrowanie i monitorowanie							A10

Źródło: Opracowanie własne.

Analizując tabelę 2, dokonano interpretacji wyników badań. Pozwoliło to na ustalenie, iż na przestrzeni lat zmieniło się postrzeganie istotności zagrożeń i obecnie najczęściej występujące to tzw. wstrzyknięcia złośliwego oprogramowania, skrypty międzyserwisowe (XSS) oraz niepoprawna obsługa uwierzytelnienia i sesji. Dodatkowo zaobserwowano powtarzalność niektórych zagrożeń na przełomie lat w badanych okresie 2003-2017 (rys. 1).



Rys. 1. Powtarzalność zagrożeń bezpieczeństwa aplikacji internetowych na przełomie lat 2003-2017

Źródło: Opracowanie własne.

Można zauważyć, iż „wstrzyknięcia" zostały odnotowane jako zagrożenie cybernetyczne już w 2003 r., a ich występowanie rosło, by od 2010 r. stanowić najczęstsze zagrożenie według OWASP. Kolejnym zagrożeniem zdefiniowanym już w 2003 r. są skrypty międzyserwisowe (XSS), będące początkowo bardzo częstym zagrożeniem, które na przestrzeni czasu zmniejszało się, choć nadal pozostało na liście. Ostatnim jest niepoprawna obsługa uwierzytelnienia i sesji, która zwykle utrzymywała się w pierwszej trójce najczęstszych zagrożeń. Jedynie w 2007 r. liczba wystąpień tego ataku zmniejszyła się i jego pozycja spadła na siódme miejsce.

4. Przegląd zagrożeń i rekomendacji zabezpieczeń aplikacji internetowych

Badania prowadzone w oparciu o zestawienia OWASP z 2017 r. wskazują na występowanie różnic w aktywności w zakresie ataków cybernetycznych w porównaniu z poprzednimi latami. Zapewnienie bezpieczeństwa w systemach informatycznych jest związane ze stałym monitorowaniem działania aplikacji internetowych pod kątem ataków skierowanych przeciw poprawnemu ich działaniu. Zapewnienie bezpieczeństwa aplikacji jest związane ze stosowaniem metod w różnych obszarach, zarówno w zarządzaniu danymi, jak i użytkowaniu API. Najnowszą wersję zaleceń OWASP związanych z projektowaniem i użytkowaniem aplikacji internetowych otwierają rekomendacje na temat ataku związanego z tzw. Injection, czyli błędach aplikacji związanych z „wstrzyknięciem". Przykładem ataku tego rodzaju jest najbardziej popularne obecnie SQL Injection, które występuje zazwyczaj w witrynach opartych na PHP i ASP, co jest spowodowane charakterystyką interfejsów komunikacji SQL tych technologii. Atak bazuje na wstrzykiwaniu tzw. wywołań SQL za pomocą danych wejściowych wybranej aplikacji. Wykonany z sukcesem pozwala na dodawanie, usuwanie i przeglądanie danych (często zawierających adresy, hasła i dane użytkowników) [Hollosi, 2013]. Czasem możliwe jest wykonanie operacji administracyjnych, takich jak wyłączenie systemu bazodanowego czy dostęp do linii poleceń systemu operacyjnego. Tego typu zagrożenie może mieć miejsce, gdy witryna przekazuje zapytania do bazy danych z niezaufanego źródła bądź też wykorzystuje je do dynamicznego ich tworzenia [Clarke, 2015].

Drugim zagrożeniem jest niepoprawna obsługa uwierzytelnienia i sesji. Występuje, gdy funkcje związane z uwierzytelnieniem i zarządzaniem sesjami nie są odpowiednio zaimplementowane. Może to spowodować dostęp do haseł, kluczy, tokenów sesji lub wykonania poleceń na prawach zalogowanego użytkownika. Błąd ten wynika z użycia protokołu HTTP, a nie HTPS (wersja szyfrowana protokołu HTTP). Możliwe jest, iż przesyłając link ze strony HTTP i otwierając go na innym komputerze, osoba niepowołana może być zalogowana na dane tego użytkownika [Mirdha, 2017].

Trzecie miejsce w rankingu zajmuje ekspozycja danych wrażliwych. Najbardziej znanym przykładem tego błędu jest przesyłanie strony logowania do poczty elektronicznej za pomocą protokołu HTTP i ograniczenie szyfrowania tylko do przesyłania loginu i hasła. Wtedy atakujący – znajdujący się między użytkownikiem a serwerem – jest w stanie zmusić użytkownika do wysłania danych najpierw do niego, a dopiero później do serwisu. Takie podejście zapew-

nia ochronę danych jedynie przed pasywnym atakującym. W przypadku aktywnego ataku ofiara nie jest w stanie wykryć ataku, jeżeli nie będzie patrzeć na kod strony [*Top 10-2017 Top 10*, b.r.].

Czwartym zagrożeniem są zewnętrzne pliki XML. Napastnikiem może być każdy, kto ma dostęp do stron lub usług sieciowych (w szczególności usług SOAP) przetwarzających kod XML. Domyślnie wiele starszych usług przetwarzających XML pozwala na określenie obiektów zewnętrznych, kodu URI, który zostaje później przetworzony podczas przetwarzania pliku XML. Zagrożenie XXE może zostać użyte do ekstrakcji danych, wykonania zdalnego polecenia na serwerze, skanowanie systemów wewnętrznych i innych rodzajów ataków. Aplikacje, a zwłaszcza usługi sieciowe oparte na XML, mogą być podatne na atak, jeżeli akceptują kod XML z niezaufanych źródeł lub wklejają niezaufane treści w dokumenty XML.

Nieprawidłowa kontrola dostępu stanowi piąte miejsce w rankingu i zawiera wszystkie zagrożenia związane ze sprawdzeniem praw dostępu do treści i funkcji aplikacji. Najczęściej spotykanym błędem jest możliwość dostania się do określonej informacji lub możliwość wywołania funkcji aplikacji przez bezpośrednie wywołani URL [*Top 10-2017 Top 10*, b.r.].

Do problemów złej konfiguracji (szóste miejsce) można zaliczyć zostawienie domyślnych haseł, konfiguracji, możliwości listowania katalogów, brak podejmowania akcji w przypadków plików z rozszerzeniem .inc, włączone raportowanie informacji odnośnie błędów aplikacji itp.

Siódme miejsce w rankingu zajmują skrypty międzyserwisowe. Błędy tego typu powstają podczas próby wyświetlenia w przeglądarce internetowej danych pochodzących od użytkownika bez wcześniejszej walidacji. Umożliwia to atakującemu wykonanie skryptu na prawach ofiary. Efektami mogą być przechwycenie sesji zalogowanego użytkownika, podmiana zawartości części serwisu czy przekierowanie użytkownika na stronę zawierającą inne szkodliwe skrypty lub oprogramowanie. Można wyróżnić trzy typy skryptów: XSS odbity (Reflected XSS), XSS magazynowany (Stored XSS) i zawierający eksploity (Sploit-pack). W pierwszym z nich atakujący przygotowuje specjalny link zawierający w sobie informacje modyfikujące oryginalną zawartość witryny (np. skrypt, który zostanie wykonany przez ofiarę). Kolejnym krokiem jest namówienie ofiary, by otworzyła za jego pomocą stronę WWW. Odbywa się to na zasadzie odbicia – osoba przy pomocy linka wprowadza skrypt, który zostaje zwrócony przez serwer i następnie wykonany przez przeglądarkę ofiary.

W XSS magazynowanym kod przekazywany jest przez ofiarę w parametrach wywołania strony. Kod atakującego pobierany jest z bazy danych w po-

dobny sposób, jak w treści dyskusji na forach czy wpisach gości. Miejsce przechowywania złośliwych informacji nie narusza sposobu obsługi przez serwer, dlatego bez problemu trafia do ofiary. Błąd polega na tym, że te mogące zostać zinterpretowane przez przeglądarkę jako zawartość witryny nie powinny być dopuszczone do zachowania i przekazania klientowi.

Sploit-pack zawierają eksploity, które są zorientowane na konkretne zagrożenia w różnych przeglądarkach. Ofiara zostaje poddana atakowi poprzez zainfekowaną stronę za pomocą zainfekowanej przeglądarki. W wyniku ataku możliwe jest zainstalowanie na komputerze ofiary złośliwego oprogramowania [*Top 10-2017 Top 10*, b.r.].

Ósmym zagrożeniem jest niezabezpieczona deserializacja. Aplikacje rozproszone, a także takie, które przechowują swój stan w systemie plików lub po stronie klienta, mogą używać serializacji obiektów [*Serializacja i deserializacja*, 2017]. Aplikacje rozproszone z publicznym dostępem oraz aplikacje, które wymagają od klienta przechowywania swojego stanu, są często narażone na manipulację serializowanych danych. Aplikacje i API są podatne, gdy mechanizm serializacji pozwala na tworzenie dowolnych typów danych. Dzieje się tak również dla aplikacji, dla których są dostępne klasy, które można połączyć ze sobą, by zmienić sposób zachowania aplikacji podczas serializacji lub już po jej wykonaniu. Dodatkowo aplikacja lub API przyjmuje i przeprowadza deserializację obiektów wgranych przez napastnika bądź aplikacja wysyła informacje o zabezpieczeniach do klienta bez jego uprzedniego uwierzytelnienia [*OWASP Top 10 2017*…, b.r.].

Używanie komponentów ze znanymi lukami w zabezpieczeniach zwykle wynika z niewiedzy użytkownika lub zaniedbań administratora. Skutkiem może być używanie przez serwer strony WWW przestarzałego systemu operacyjnego pełnego luk i dobrze znanych błędów.

Ostatnim zagrożeniem na liście z listopada 2017 r. są niewystarczająca archiwizacja i monitorowanie. Brak dostatecznej archiwizacji dotyczy systemów, w których informacje o istotnych zdarzeniach (zalogowanie się użytkownika, nieudane próby logowania, transakcje o wysokiej wartości) nie zostają zachowywane. Logi aplikacji i API nie są monitorowane pod względem podejrzanej aktywności lub progi powiadomień o pojawiających się zagrożeniach i metody ich zwalczania są nieskuteczne lub nie zostały zaimplementowane. Dla dużych organizacji wielkim ryzykiem byłby brak aktywnego reagowania na zagrożenia (brak powiadomień w czasie rzeczywistym o zagrożeniach, brak mechanizmów zwalczania zagrożeń), jak np. blokowanie zautomatyzowanych ataków na aplikacje sieciowe lub API. Reakcja na zagrożenie nie musi być dostrzegalna dla

samego napastnika – wystarczy, że aplikacja i związana z nią infrastruktura są w stanie wykryć zagrożenie i powiadomić o nim człowieka lub same użyć odpowiednich mechanizmów obronnych [Beal, b.r.].

Opierając się na najnowszych danych opublikowanych przez OWASP, przeprowadzono porównanie listy dziesięciu najczęstszych zagrożeń aplikacji internetowych z rankingów opublikowanych w kwietniu i listopadzie 2017 r. oraz zamieszczono je w tabeli 3. Zmienność występowania zagrożeń i ich różnorodność potwierdzają tezę o trudności zapewnienia bezpieczeństwa cybernetycznego w aplikacjach internetowych. W pierwszej edycji zauważono dwa nowe zagrożenia, tj. niewystarczającą ochronę przed atakami i niechronione interfejsy API. W kolejnej edycji pominięto te zagrożenia i ustalono trzy inne: zewnętrzne pliki XML, niezabezpieczoną deserializację oraz niewystarczające rejestrowanie i monitorowanie.

Tabela 3. Porównanie najczęściej występujących zagrożeń aplikacji internetowych w kwietniu (RC1) i listopadzie (RC2) 2017 r. według OWASP

Czynniki zagrożeń aplikacji internetowych (kwiecień 2017 r.)	**Czynniki zagrożeń aplikacji internetowych (listopad 2017 r.)**
A1. „Wstrzyknięcie”	A1. „Wstrzyknięcie”
A2. Niepoprawna obsługa uwierzytelnienia i sesji	A2. Niepoprawna obsługa uwierzytelnienia i sesji
A3. Skrypty między serwisowe (XSS)	A3. Ujawnienie danych wrażliwych
A4. Nieprawidłowa kontrola dostępu	A4. Zewnętrzne pliki XML (XXE)
A5. Błędy w konfiguracji zabezpieczeń	A5. Nieprawidłowa kontrola dostępu
A6. Ujawnienie danych wrażliwych	A6. Błędy w konfiguracji zabezpieczeń
A7. Niewystarczająca ochrona przed atakami	A7. Skrypty między serwisowe (XSS)
A8. Skrypty fałszujące dostęp do witryny (XSRF)	A8. Niezabezpieczona deserializacja
A9. Używanie komponentów ze znanymi lukami w zabezpieczeniach	A9. Używanie komponentów ze znanymi lukami w zabezpieczeniach
A10. Niechronione interfejsy API	A10. Niewystarczające rejestrowanie i monitorowanie

Źródło: Opracowanie własne na podstawie danych OWASP [*OWASP Top 10 2017*…, b.r.; *Top 10-2017 Top 10*, b.r.]; Krysiak [2018, s. 162].

Pierwsza edycja z kwietnia 2017 r. (RC1) zawierała dwa już później nieuwzględnione zagrożenia. Według OWASP niedostateczna ochrona przed atakami jest równoznaczna z brakiem odpowiedniego zabezpieczenia kanałów przesyłu danych. Spowodowane może być brakiem SSL podczas logowania, wygasłymi certyfikatami czy słabym szyfrowaniem połączeń. Zgodnie z rekomendacjami organizacji każdy użytkownik, który może wysyłać żądania do API, jest potencjalnym napastnikiem. Podobnie jak tradycyjna aplikacja, API także jest narażone na „wstrzyknięcia” kodu, przejęcie kontroli, szyfrowania czy konfiguracji.

Podsumowanie

Różnorodność rodzajów i wersji ataków cybernetycznych stwarza ogromne zagrożenia bezpieczeństwa w zakresie zarządzania danymi i informacjami w aplikacjach internetowych, przede wszystkim ze względu na różnorodne technologie tworzenia aplikacji internetowych, niejednokrotnie niewspieranych deweloperskо. Warto zauważyć, że wraz z rozwojem aplikacji internetowych i infrastruktury informatycznej nieustannie zmieniają się warunki popełnienia przestępstwa cybernetycznych przez hakerów. Organizacje zajmujące się monitorowaniem zagrożeń oraz wdrażania i zastosowaniem odpowiednich metod zabezpieczeń w aplikacjach internetowych – tworzące OWASP – stanowią podstawowe źródło wiedzy dla dalszych analiz związanych z badaniem podatności aplikacji internetowych.

W artykule dokonano analizy występowania różnych ataków cybernetycznych, odnosząc się do cyklicznie wyznaczanych rankingów dziesięciu najczęściej występujących podatności aplikacji internetowych, opublikowanych podczas minionych 15 lat na świecie. Zaproponowano rekomendacje w zakresie obsługi aplikacji internetowych umożliwiające ograniczenie wystąpienia określonej podatności, uwzględniając różne dostępne źródła wiedzy na temat ataków cybernetycznych oraz metody i główne zalecenia stowarzyszenia OWASP. Ochrona danych w sieci związanych z obsługą transakcji realizowanych w Internecie stanowi podstawowy czynnik zapewniający bezpieczeństwo systemów informatycznych. Różnorodność ataków cybernetycznych stwarza stale zagrożenie bezpieczeństwa aplikacji internetowych, a w konsekwencji może przyczynić się do ryzyka wystąpienia naruszenia prywatności, poufności i integralności danych i informacji wykorzystywanych do realizacji procesów zarówno w organizacjach gospodarczych, jak i administracji. Rozwój technologii internetowych oraz koncepcje wykorzystania aplikacji internetowych m.in. do obsługi Internetu rzeczy w inteligentnym otoczeniu [Szpor, red., 2015] oraz gospodarce 4.0 [Sołtysik-Piorunkiewicz, Krysiak, 2020] będą wymagały stałego monitorowania i zbadania nowych podatności związanych z zarządzaniem ryzykiem bezpieczeństwa aplikacji i sieci, a także odniesienia założeń inżynierii bezpieczeństwa systemów informatycznych do aspektów związanych z zapewnieniem prywatności ich użytkowników oraz możliwości zabezpieczenia użytkowników przed zagrożeniami wynikającymi ze stosowania technologii mobilnych [Sołtysik-Piorunkiewicz, 2018].

Literatura

About The Open Web Application Security Project, OWASP, https://www.owasp.org/index.php/About_The_Open_Web_Application_Security_Project (dostęp: 20.03.2019).

About Us, ISACA, www.isaca.org/about-isaca/Pages/default.aspx (dostęp: 18.03.2019).

Beal V., *API – Application Program Interface*, Webopedia, https://www.webopedia.com/TERM/A/API.html (dostęp: 30.03.2019).

Clarke J. (2015), *SQL Injection Attack and Defence*, Elsevier, United States.

Czym są certyfikaty SSL, Certum by Asseco, https://ssl.certum.pl/certyfikaty/certy, informacje_co_to_jest_certyfikat_ssl.xml (dostęp: 31.03.2019).

Exploit, Dobry Słownik, https://dobryslownik.pl/slowo/exploit/221932/0/234770/ (dostęp: 30.03.2019).

Hollosi A. (2013), *Integracja PHP z Windows: optymalna wydajność i bezpieczeństwo*, Helion, Gliwice.

Konieczny P. (2009), *10 najpopularniejszych błędów w webaplikacjach*, Niebezpiecznik, 4 grudnia, https://niebezpiecznik.pl/post/10-najpopularniejszych-bledow-w-webaplikacjach/ (dostęp: 19.03.2019).

Krysiak M. (2018), *Ochrona przed najczęstszymi zagrożeniami aplikacji internetowych na podstawie badań OWASP* [w:] R. Kruzel, R. Balina, H. Tańska, S. Ejdys, M. Drewniak (red.), *Ludzie nauki w kręgu interdyscyplinarnych badań*, INTELLECT, Waleńczów, s. 160-168.

Mirdha R. (2017), *Learn Hacking on Web Application from Beginner to Advanced*, India.

OWASP 2007 Top 10 Presentation (2009), Bretthard, 21th October, http://bretthard.in/post/owasp-2007-top-10-presentation (dostęp: 20.03.2019).

OWASP Top 10 – 2013. The Ten Most Critical Web Application Security Risks, OWASP, https://www.owasp.org/images/f/f8/OWASP_Top_10_-_2013.pdf (dostęp: 20.03.2019).

OWASP Top 10 2017. The Ten Most Critical Web Application Security Risks, OWASP, https://www.owasp.org/images/b/b0/OWASP_Top_10_2017_RC2_Final.pdf (dostęp: 20.03.2019).

OWASP Top Ten, OWASP, https://www.owasp.org/index.php/2004_Updates_OWASP_Top_Ten_Project (dostęp: 20.03.2019).

Rozszerzenie pliku .INC, Download Astro, http://pl.downloadastro.com/Pliki%20Windows/inc/ (dostęp: 30.03.2019).

Sejdak K. (2013), *Co oferuje nam OWASP?*, WebSecurity, 11 stycznia, http://websecurity.pl/co-oferuje-nam-owasp/ (dostęp: 18.03.2019).

Serializacja i deserializacja (2017), Microsoft, 30 marca, https://msdn.microsoft.com/pl-pl/library/ms731073(v=vs.110).aspx (dostęp: 30.03.2019).

Sołtysik-Piorunkiewicz A. (2018), *Modele oceny użyteczności i akceptacji mobilnych systemów zarządzania wiedzą o zdrowiu*, Uniwersytet Ekonomiczny, Katowice.

Sołtysik-Piorunkiewicz A., Krysiak M. (2020), *The Cyber Threats Analysis for Web Applications Security in Industry 4.0* [w:] M. Hernes et al. (eds.), *Towards Industry 4.0 – Current Challenges in Information Systems*, Springer, s. 127-141.

Szpor G., red. (2015), *Internet rzeczy. Bezpieczeństwo w Smart City*, C.H.Beck, Warszawa.

Top 10-2017 Top 10, OWASP, https://www.owasp.org/index.php/Top_10_2017-Top_10 (dostęp: 20.03.2019).

Wichers D., *OWASP Top 10 – 2010. The Top 10 Most Critical Web Application Security Risks*, OWASP, https://www.owasp.org/images/6/67/OWASP_AppSec_Research_2010_OWASP_Top_10_by_Wichers.pdf (dostęp: 20.03.2019).

SECURITY ENGINEERING OF WEB APPLICATIONS BASED ON THREAT ANALYSIS AND OWASP RECOMMENDATIONS

Summary: The article presents the security threats of web applications in the design and development of information systems based on the guidelines resulting from software security engineering. The article identifies the variety and variability of security threats for web applications. The purpose of the presented research is to analyze the trend in the appearance of threats of web applications which are based on data collected by the OWASP Foundation published over the years 2003-2017. The first chapter of the article presents the role and tasks of the OWASP Foundation against guidelines developed by organizations dealing with the security of web applications. The second chapter describes the most common security threats of web applications. The third chapter analyses the occurrence and compares the frequency of threats to Internet applications in the years 2003-2017, and finally, the fourth chapter presents the ways of protecting web applications and recommendations to be taken into consideration in the design and development of IT systems using web applications and during their usage.

Keywords: security engineering, security of web applications, threats and vulnerabilities, data and information security, OWASP.