EduAkcja. Magazyn edukacji elektronicznej
nr 1 (9)/2015, str. 80—89

Wyzwania bezpieczeństwa nowoczesnych platform nauczania zdalnego

Paweł Lubomski
Politechnika Gdańska
lubomski@pg.gda.pl

Streszczenie: W artykule zaprezentowano aspekty bezpieczeństwa nowoczesnych platform nauczania zdalnego. Przedstawiono ich charakterystykę i wyzwania technologiczne. Zdefiniowano bezpieczeństwo i istniejące w tym obszarze zagrożenia. Przybliżono metody oceny poziomu bezpieczeństwa. Na bazie wdrożonej na Politechnice Gdańskiej platformy eNauczanie PG omówiono sposoby zapewniania zakładanego poziomu bezpieczeństwa takich systemów.

Słowa kluczowe: e-learning, bezpieczeństwo, interoperacyjność, audyt bezpieczeństwa, analiza ryzyka, niezawodność, wydajność

1. Wprowadzenie

Budowa bezpiecznego systemu internetowego jest w obecnych czasach niemałym wyzwaniem, szczególnie, gdy jest to system dostępny powszechnie w Internecie. A właśnie takie są platformy nauczania zdalnego. Związane jest to przeważnie z ofertą takiej platformy skierowaną do szerokiego grona odbiorców.

Można rozróżnić dwa kierunki rozwoju współczesnych systemów e-learningowych: nastawione na zamknięte kursy dla określonych, dobrze znanych odbiorców oraz drugie, publiczne, skierowane do ogółu społeczeństwa, często anonimowego. Pierwsze z nich to platformy realizacji zajęć dydaktycznych wspierających (blended-learning (Thorne, 2003)) lub realizujących w całości zdalnie program przedmiotu szkolnego lub uczelni wyższej. Coraz częściej w takiej postaci realizowane są również różnego typu komercyjne kursy doskonalące. Rozwój drugiego nurtu kursów jest efektem licznych programów i projektów mających na celu zmniejszenie wykluczenia technologicznego społeczeństwa, wyrównanie szans osób pochodzących ze środowisk miejskich i wiejskich, osób niepełnosprawnych, itp. Są to przeważnie programy finansowane ze środków Unii Europejskiej. Nurt ten będzie odgrywał bardzo istotną rolę szczególnie w programach uruchamianych w perspektywie finansowej 2014–2020 (np. Program Operacyjny Wiedza Edukacja Rozwój).

Jednak niezależnie od nurtu, w ramach którego będą uruchamiane platformy, należy oczekiwać, że korzystać z nich będzie bardzo liczne grono użytkowników. Dodatkowo metodycy nauczania zdalnego doskonalą swój warsztat wprowadzając kolejne nowinki technologiczne w proces dydaktyczny.

Przygotowanie dobrego kursu na platformę e-learningową wymaga wiele wysiłku i poświęconego czasu. Przekłada się to na wartość samego kursu, a więc stanowi dzieło autorskie, które należy chronić przed nielegalnym i niekontrolowanym powielaniem i wykorzystywaniem.

Wszystkie wspomniane wyżej aspekty rozwoju współczesnych platform nauczania zdalnego powodują, że realizując liczne cele merytoryczne nie możemy pomijać bardzo istotnych aspektów technologicznych mających wpływ na szeroko rozumiane bezpieczeństwo samego systemu, jak i danych przez niego przetwarzanych.

2. Charakterystyka technologiczna

Technologia, w jakiej są wykonane omawiane platformy nauczania zdalnego, jest zdeterminowana szeroką dostępnością tychże. Są to zatem praktycznie w stu procentach platformy webowe (systemy internetowe). Można uzyskać do nich dostęp o dowolnej porze. Dostęp do systemu nie wymaga przeważnie żadnego specjalizowanego oprogramowania – wystarczy dowolna przeglądarka WWW. Interfejsy użytkownika budowane są z intensywnym wykorzystaniem JavaScriptu, technologii AJAX, rozwiniętym CSS oraz zyskującymi coraz szersze uznanie elementami standardu HTML 5.

Takie podejście umożliwia dostępność kursów nauczania zdalnego na dowolnym komputerze z dowolnym systemem operacyjnym oraz na większości urządzeń mobilnych, które zyskują coraz większą popularność i sukcesywnie wypierają tradycyjne komputery osobiste. Jest to o tyle istotne, że rozwiązania te nie są ze sobą zgodne technologicznie i często luźno interpretują standardy.

Nie bez znaczenia jest fakt, że zastosowanie takiej architektury przenosi obciążenie obliczeniowe na serwery znajdujące się w chmurze (Mather et al., 2009) . Dzięki temu narzędzia działają sprawnie nawet na urządzeniach o mniejszych zdolnościach obliczeniowych (starsze komputery, smartfony i tablety).

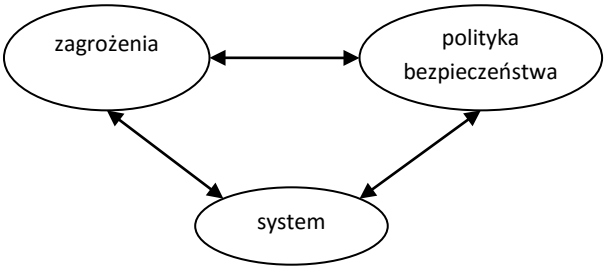
Stosowanie wielu rozwiązań technologicznych wynikające z łączenia różnego typu treści (tekst, dźwięk, obraz, film) wiąże się z dużym rozproszeniem źródeł tych danych oraz wymaganą interoperacyjnością ich dostawców. Przykładem może być w tym miejscu serwowanie filmów z usługi YouTube, połączone z interakcją oprogramowaną w usłudze specjalizującą się w tego typu zadaniach (Zaption, 2015) oraz samym kursem serwowanym na platformie. Innym przykładem może być usługa realizacji streamingu wideo dla wideo-konsultacji lub wykładu on-line również zintegrowana z platformą (Cisco WebEx, 2015).

W ten sposób dochodzimy do ostatniego elementu charakterystycznego dla tego typu rozwiązań: architektury usługowej (ang. Service Oriented Architecture – SOA) (Erl, 2008). To właśnie umiejętne połączenie wielu różnych usług pozwala stworzyć atrakcyjne i wartościowe pod względem merytorycznym narzędzie do nauki zdalnej. Możliwość „porozumienia się" poszczególnych usług, często różnych technologicznie, jest możliwa dzięki opracowanym standardom interoperacyjności (Krawczyk i Lubomski, 2010)(Liu i Hoang, 1994).

3. Bezpieczeństwo

Bezpieczeństwo systemów internetowych jest bardzo rozległym zagadnieniem. Najczęściej w literaturze spotyka się definiowanie go przez pięć głównych atrybutów: identyfikację i uwierzytelnianie, kontrolę dostępu do zasobów, zabezpieczenie danych i komunikację (poufność i integralność), niezaprzeczalność oraz dostępność systemu (Anisetti et al., 2013). Mówi się więc o systemie działającym w sposób przewidywalny i zgodnie z oczekiwaniami użytkownika.

Warto wyróżnić dwa główne obszary bezpieczeństwa: kontrola dostępu do zasobów (ang. *security*) oraz zapewnienie dostępności i ciągłości działania (ang. *safety*). Obszary te są komplementarne i w pewnym zakresie oddziałują na siebie. Rys. 1 przedstawia otoczenie, w którym działa system.



Rysunek 1. Środowisko działania systemu

Jak widać na tym rysunku, system realizuje pewną przyjętą politykę bezpieczeństwa. Polityka bezpieczeństwa z perspektywy kontroli dostępu jest zbiorem reguł wyznaczanych przez organizację, które musi spełniać użytkownik otrzymujący dostęp do zasobu. Wyróżnia się 4 podstawowe typy polityk kontroli dostępu: ABAC (ang. Attribute-Based Access Control), DAC (ang. Discretionary Access Control), MAC (ang. Mandatory Access Control) i RBAC (ang. Role-Based Access Control) (Benantar, 2006).

Realizacja takiej polityki odbywa się poprzez implementację i wykorzystanie odpowiednich mechanizmów bezpieczeństwa. Mechanizmy kontroli dostępu odpowiadają metodom, narzędziom lub procedurom, z wykorzystaniem których implementowana jest polityka bezpieczeństwa (Benantar, 2006). Mechanizmy takie mogą być zautomatyzowane, w pełni zaszyte w system (najlepszy wariant), mogą być również częściowo lub całkowicie manualne, poza systemem (w postaci procedur postępowania) – w takim przypadku realizowane przez fizyczne osoby, np. administratora systemu. Wybór zastosowanych mechanizmów w znacznym stopniu zależy od specyfiki systemu. Inne będą stosowane dla systemów internetowych, a inne np. dla urządzeń mobilnych (Leavitt, 2011)(Hansen i Oleshchuk, 2003)(Kirkpatrick, Bertino, 2010). Stosuje się mechanizmy tzw. piaskownicy (ang. *sandbox*) lub warstwowej kontroli zgodnej z modelem DAC (Miller, 2011).

Mechanizmy te powstają jako reakcja na pojawiające się zagrożenia (ang. *threats*). Zagrożenia dla bezpieczeństwa systemów komputerowych ciągle ewoluują. Pojawiają się również nowe. Produkowane oprogramowanie jest wytwarzane coraz szybciej kosztem obniżenia jego jakości. Dodatkowy wpływ na to ma rosnący poziom skomplikowania. Przekłada się to na rosnącą liczbę wprowadzonych nieświadomie podatności (ang. *vulnerability*). Ponieważ dane zawarte w systemach stanowią coraz większą wartość, pojawia się wiele exploitów – programów wykorzystujących te podatności celem przełamania zabezpieczeń. Zaobserwować można pewnego rodzaju wyścig pomiędzy osobami szukającymi nowych podatności a zespołami łatającymi dziury. Okoliczności te powodują, że sytuacja jest bardzo zmienna. Doskonałym przykładem jest liczba codziennie identyfikowanych nowych wirusów komputerowych. Aby zapanować nad tak dynamicznie zmienną sytuacją, potrzebne jest prowadzenie odpowiednio skategoryzowanej ewidencji wykrytych zagrożeń. Do tego celu różne instytucje zajmujące się omawianym zagadnieniem tworzą swoje klasyfikacje. Najpopularniejsze z nich to:

- US-CERT Vulnerability Notes (Software Engineering Institute, US-CERT Vulnerability Notes, 2014),
- MITRE Common Vulnerabilities and Exposures (CVE) (CVE Community, Common Vulnerabilities and Exposures, 2014),
- MITRE Common Weakness Enumeration (CWE) (CWE Community, Common Weakness Enumeration, 2014),

- National Vulnerability Database (NVD) (National Vulnerability Database Version 2.2, 2014),
- SecurityFocus Vulnerability Database and BugTraq mail list (SecurityFocus Vulnerability Database and BugTraq mail list, 2014),
- Open Source Vulnerability Database (OVSDB) (Open Sourced Vulnerability Database, 2014),
- The Web Application Security Threat Classification (The WASC Threat Classification v2.0, 2014),
- The Open Web Application Security Project (OWASP) (OWASP community. The Open Web Application Security Project (OWASP), 2014),
- biuletyny bezpieczeństwa dostawców oprogramowania.

Ze względu na powszechną dostępność systemów internetowych zorientowanych na usługi (a takimi właśnie są najczęściej platformy nauczania zdalnego) dotyczą ich wszystkie typy zagrożeń i posiadają bardzo dużo potencjalnych podatności. W związku z tym praktycznie dowolna z powyższych klasyfikacja zagrożeń będzie dobrą podstawą do analizy zagrożeń systemu.

4. Ocena bezpieczeństwa

Ocena poziomu bezpieczeństwa systemu jest niezwykle złożonym i trudnym zagadnieniem. Główna trudność polega na doborze odpowiednich metryk (Hinson, 2006, Hauser i Katz, 1998). Dodatkowa trudność wynika z pytania: jak zmierzyć brak incydentów? (Payne, 2006). Jednym z podejść jest metoda audytów, które sprawdzają kolejno z listą zgodność z dobrymi praktykami (ang. *best practices*) w zakresie bezpieczeństwa. Przykładem takich norm są:

- brytyjska BS 7799-2,
- międzynarodowa ISO/IEC 27001.

Dodatkowo audyty mogą zawierać część techniczną, w skład której wchodzą testy penetracyjne. Bazują one na listach znanych typów podatności i próbie ich odnalezienia w testowanym systemie.

Każda wykryta podatność powinna być analizowana pod kątem ryzyka, jakie niesie jej wykorzystanie przez potencjalnego atakującego (Lund et al., 2010, Mather et al., 2009) . Na podstawie tak przygotowanego i na bieżąco aktualizowanego zestawienia określa się priorytety w usuwaniu podatności (Damián-Reyes et al., 2009, Dimmock, 2004). Warto zwrócić uwagę, że już na etapie tworzenia polityki bezpieczeństwa powinna zostać przeprowadzona analiza ryzyka wraz z określeniem progu akceptowalnego jego poziomu (Disterer, 2013, Clinch, 2009).

Powstało wiele metod oraz technik analizy i oceny ryzyka. Warto wymienić cztery:

- Microsoft Security Response Center Security Bulletin Severity Rating System (Microsoft Security Response Center Security Bulletin Severity Rating System, 2002),
- US-CERT Vulnerability Metric (US-CERT Vulnerability Metric, 2014),
- Common Vulnerability Scoring System (CVSS) (NVD Common Vulnerability Scoring System Support v2., 2007)(Mell et al., 2007)(Mell et al., 2006),
- SANS Critical Vulnerability Analysis Scale (Mell et al., 2006).

Obecnie rozwijany jest głównie CVSS. Analizę przeprowadza się zgodnie z wytycznymi Common Vulnerability Scoring System Version 2.0 (Mell et al., 2014) z wykorzystaniem kalkulatora dostarczonego przez National Institute of Standards and Technology (Common Vulnerability Scoring System Version 2 Calculator, 2014).

CVSS v2 został stworzony do oceny krytyczności poszczególnych podatności, tak aby możliwe było ich wzajemne porównanie ze sobą i właściwa priorytetyzacja usuwania. Składa się on z 3 grup metryk:

- bazowych określających podstawowe cechy charakterystyczne dla podatności, które są niezmienne w czasie i niezależne od środowiska użytkownika/organizacji,
- czasowych – ich wartość zmienia się w czasie życia podatności od czasu jej ujawnienia do

zastosowania zabezpieczeń,
- środowiskowych, które zależą od środowiska konkretnego użytkownika/organizacji (np. straty ekonomiczne wynikające z utraty produktywności lub wizerunku).

Metryki te służą do wyznaczenia numerycznej wartości oraz tzw. wektora CVSS v2, które wskazują na krytyczność podatności (luki).

5. Platforma eNauczanie PG

Rozwijana na Politechnice Gdańskiej platforma nauczania zdalnego eNauczanie PG (eNauczanie PG, 2014, Lubomski i Żuchowski, 2014) jest doskonałym przykładem analizowanych systemów internetowych zorientowanych na usługi i służących jako narzędzie e-learningu. Łączy ona w sobie wiele różnych komponentów, w większości dostępnych na licencjach open source. Trzonem platformy jest system zarządzania nauczaniem (ang. *learning management system*). Politechnika Gdańska używa oprogramowania Moodle (Moodle, 2014).

Platforma eNauczanie PG jest silnie zintegrowana z innymi systemami uczelni. Podstawowym punktem styku jest wyniesiony poza platformę, wspólny dla wielu systemów PG, Centralny Punkt Logowania (Krawczyk i Lubomski, 2010). Takie rozwiązanie architektoniczne ma kilka zalet. Po pierwsze użytkownik nie musi pamiętać wielu różnych haseł do każdego systemu oddzielnie. Ma to szczególnie istotne znaczenie w dużej organizacji, w której pracuje dużo pracowników, którzy są w znaczącym stopniu wspierani przez różne systemy informatyczne. Pamiętanie różnych haseł (bo często w różnych systemach stosowana jest różna polityka haseł) do każdego systemu oddzielnie rodzi wiele problemów, szczególnie, gdy nie ze wszystkich systemów korzysta się codziennie. Wbrew pozorom w takim rozwiązaniu wzrasta poziom bezpieczeństwa systemów, ponieważ jedno hasło można zapamiętać i nie trzeba zapisywać, co często się zdarza, gdy tych haseł jest więcej.

Drugą zaletą takiego rozwiązania jest skupienie w jednym systemie tego niezwykle krytycznego pod kątem bezpieczeństwa procesu, jakim jest proces uwierzytelniania. Dzięki takiemu podejściu dane uwierzytelniające (najczęściej login i hasło) są podawane, przetwarzane i przechowywane tylko w jednym miejscu, co zmniejsza ryzyko ich wycieku w przypadku błędów w implementacji w różnych systemach – wystarczy w tym przypadku skupić się na jednym systemie, o bezpieczeństwo którego należy szczególnie zadbać.

Kolejną bardzo istotną zaletą takiego rozwiązania (szczególnie w kontekście systemów łączących różne technologie) jest możliwość zastosowania mechanizmu jednokrotnego logowania (ang. *Single Sign On*) (Krawczyk i Lubomski, 2010, Goth, 2008, De Capitani Di Vimercati et al., 2012). Zastosowanie takiego mechanizmu powoduje, że użytkownik swoje dane uwierzytelniające podaje tylko pierwszy raz, kiedy jeden z systemów, z których korzysta, będzie wymagał uwierzytelnienia. Każde kolejne żądanie systemu (nieważne którego) do uwierzytelnienia się zrealizowane zostanie w sposób niezauważalny dla użytkownika – system w tle skomunikuje się z Centralnym Punktem Logowania PG i uzyska dane osoby zalogowanej. Jest to więc przede wszystkim duża wygoda użytkownika – tylko raz w ramach swojej sesji musi podawać login i hasło, mimo że korzysta z wielu systemów.

Rozwiązanie to umożliwia również integrację wielu systemów, a nawet komponentów jednego systemu, będących niezależnymi usługami, działającymi w sposób niezależny od siebie. W dodatku odbywa się to w sposób przezroczysty dla użytkownika. Przy zachowaniu w miarę jednolitego interfejsu użytkownika, przechodzenie pomiędzy systemami i korzystanie z różnych usług jest dla niego praktycznie niezauważalne. Ma to szczególne znaczenie w przypadku platform nauczania zdalnego, które muszą integrować wiele różnych technologicznie usług (elementy interaktywne, video, duże pliki z danymi i treściami, fora dyskusyjne, itp.). Wszystkie te dane powinny być chronione, więc praktycznie wszystkie wymagają uwierzytelniania.

Zastosowanie jednego systemu uwierzytelniania wspólnego dla wszystkich systemów orga-

nizacji determinuje jednolitą i jednoznaczną identyfikację osób pomiędzy tymi systemami. Ponownie ma to ogromne znaczenie podczas integracji tych systemów. Dzięki temu możliwe było wykonanie sterowania z poziomu systemu obsługi dydaktyki (zlokalizowanego na Politechnice Gdańskiej na platformie MojaPG (Politechnika Gdańska, Moja PG, 2013)) listą osób zapisywanych i uprawnionych do realizacji kursu nauczania zdalnego na platformie eNauczanie PG w systemie Moodle. W ten sposób uzyskano spójność list osób uprawnionych do realizacji przedmiotu z listami osób uprawnionymi do uczestnictwa w zajęciach realizowanych w trybie e-learningu. Dodatkowo umożliwiło to zautomatyzowany przepływ danych dotyczących postępów (zaliczeń) uczestników pomiędzy tymi systemami.

W tym „dobrze poukładanym świecie" pojawił się jednak pewien problem. Platforma eNauczanie PG oprócz wspierania tradycyjnego toku nauczania, pełni również rolę narzędzia otwartego na szerokorozumianego mieszkańca regionu – część kursów uruchamianych na platformie ma być dostępna dla uczestników uniwersytetów trzeciego wieku, uczniów szkół średnich, czy zwykłego obywatela. Są to osoby spoza organizacji, jaką jest Politechnika Gdańska, więc nie będą oni mieli kont centralnych. Dlatego zdecydowano o zastosowaniu dualnego systemu uwierzytelniania – użytkownik może wybrać, czy chce zalogować się kontem politechnicznym, czy też stworzyć swoje „małe konto" tylko na potrzeby platformy eNauczanie PG. Jednak preferowane jest konto politechniczne i część platformy jest nieosiągalna z poziomu „małego konta". Wiąże się to również z tym, że w przypadku kont centralnych osoby są jednoznacznie identyfikowane – sam proces aktywacji konta i różne mechanizmy bezpieczeństwa dotyczące konta (np. potrzeba osobistego stawiennictwa i wylegitymowania się w określonych przypadkach) minimalizują szansę na wykorzystanie konta przez inne osoby. W przypadku „małych kont" do końca nie jesteśmy pewni, jakiej osoby dotyczą, ponieważ nie jest przeprowadzany proces jednoznacznej identyfikacji takiej osoby.

Platforma zlokalizowana jest na centralnych serwerach Centrum Usług Informatycznych Politechniki Gdańskiej. Serwery te pracują w technologii blade przy pełnej wirtualizacji serwerów logicznych. Dzięki takiemu rozwiązaniu możliwe jest bardziej efektywne wykorzystanie posiadanych zasobów sprzętowych. Dodatkowo możliwość dynamicznej alokacji zasobów sprzętowych (moc obliczeniowa procesora, pamięć podręczna, pamięć trwała) pozwala na szybką reakcję na zmieniające się obciążenie platformy. Odbywa się to dwutorowo: poprzez dodanie dodatkowych zasobów do już istniejących serwerów logicznych (skalowanie pionowe) lub poprzez dodanie kolejnych węzłów obliczeniowych (skalowanie poziome).

Stosowanie tego drugiego rozwiązania, mimo że trudniejsze w implementacji i pochłaniające więcej zasobów, powoduje również wzrost niezawodności działania (ang. *High Availability*). Spowodowane jest to faktem, że upadek jednego z równoległych węzłów nie powoduje upadku całej platformy, ponieważ jego obowiązki przejmują automatycznie pozostałe węzły.

6. Najważniejsze zagrożenia platformy e-Nauczenie PG

Platforma eNauczanie PG posiada bardzo wiele cech systemu internetowego. W związku z tym z tą kategorią systemów będzie dzielić najistotniejsze zagrożenia. Dlatego analizę potencjalnych zagrożeń w sferze zabezpieczenia przed nieautoryzowanym dostępem najlepiej rozpocząć od listy OWASP Top Ten (OWASP Top Ten Project, 2015). Zawiera ona 10 kategorii zagrożeń najczęściej występujących w systemach internetowych. Poniżej omówiono je skrótowo.

1. A1-Injection – obejmuje wszelkiego typu wstrzyknięcia w proces przetwarzania, takie jak SQL-, OS-, LDAP-injection. Mają miejsce, gdy odpowiednio spreparowane dane przesłane do systemu są potraktowane przez interpreter jako część polecenia lub zapytania.
2. A2-Broken Authentication and Session Management – to wszelkiego rodzaju błędy w implementacji mechanizmu uwierzytelniania lub utrzymania sesji pozwalające na przechwycenie sesji lub danych uwierzytelniających użytkowników.

3. A3-Cross-Site Scripting (XSS) – występuje, gdy system niewłaściwie waliduje dane przesyłane do systemu, w efekcie czego u innego użytkownika wyświetlającego te dane może zostać wykonany w przeglądarce WWW kod Java Script.
4. A4-Insecure Direct Object References – obejmuje błędy implementacyjne wynikające z braku kontroli dostępu na poziomie dostępu do poszczególnych obiektów na podstawie ich identyfikatorów.
5. A5-Security Misconfiguration – sprowadza się do zaniechań w aktualizacji oprogramowania i stosowania domyślnych danych konfiguracyjnych.
6. A6-Sensitive Data Exposure – to wydobycie przez atakującego wrażliwych danych (takich jak dane kart kredytowych, hasła), które powinny być przechowywane w postaci niejawnej (zaszyfrowane lub w postaci skrótów).
7. A7-Missing Function Level Access Control – polega na braku kontroli dostępu do funkcjonalności weryfikowanej po stronie serwera (funkcjonalność, mimo że niewidoczna na interfejsie, można wywołać preparując odpowiednio żądanie).
8. A8-Cross-Site Request Forgery (CSRF) – grupa ataków polegająca na takiej budowie przez atakujących stron, których odwiedzenie powoduje wywołanie żądania użytkownika do systemu w sposób niejawny, wykorzystujące jego identyfikator sesji.
9. A9-Using Components with Known Vulnerabilities – analogicznie do A5 stosowane biblioteki, frameworki i inne moduły powinny być aktualizowane do najnowszych wersji, w których usuwane są wykryte zagrożenia i błędy.
10. A10-Unvalidated Redirects and Forwards – brak odpowiedniej walidacji adresów, na które system przekierowuje użytkownika, może umożliwić przekierowanie go na strony zawierające phishing lub malware.

7. Ocena bezpieczeństwa platfromy e-Nauczanie PG

Jak już wcześniej wspomniano, jedną z metod oceny bezpieczeństwa systemu jest przeprowadzenie audytu bezpieczeństwa celem wykrycia podatności, a następnie analiza ryzyka wiążącego się z wykrytymi podatnościami. Platforma eNauczanie PG, podobnie jak pozostałe systemy centralne Politechniki Gdańskiej, podlega takim okresowym audytom bezpieczeństwa.

W ramach przeprowadzonych testów penetracyjnych wykryto 8 podatności, z których 3 sklasyfikowano na poziomie średnim i 5 na poziomie niskim. Co ciekawe, niektóre z podatności występują bezpośrednio w platformie Moodle, a więc narzędziu powszechnie wykorzystywanym przez wiele różnych instytucji. Dobrą praktyką wśród badaczy zajmujących się bezpieczeństwem jest nieudostępnianie publicznie szczegółów wykrytych luk bezpieczeństwa, tylko zgłaszanie ich bezpośrednio do autorów oprogramowania. Tak też uczyniono w powyższym przypadku.

Dla każdej wykrytej podatności została przeprowadzona analiza ryzyka, jakie ta podatność ze sobą niesie. Na tej podstawie wyznaczone zostały priorytety działań naprawczych.

Należy mieć na uwadze, że stan ten nie jest stały. Intensywny rozwój platformy może spowodować pojawienie się kolejnych, nowych podatności wynikających z błędów implementacyjnych lub nawet koncepcyjnych i architektonicznych na etapie projektowania i łączenia różnych usług. Oprócz bieżącego rozwoju, również sytuacja „po drugiej stronie" nie jest stała – ciągle odkrywane są nowe podatności oraz nowe ich rodzaje w już istniejącym oprogramowaniu. W związku z tym takie audyty należy ponawiać okresowo.

W przypadku platform nauczania zdalnego, równie ważne jak zabezpieczenie przed nieautoryzowanym dostępem (ang. *security*), jest zapewnienie niezawodności i ciągłości działania (ang. *safety*). W obszarze tym wyróżnić można dwa zagadnienia: zabezpieczenie danych przed utratą oraz odpowiednia wydajność.

Pierwsze z nich realizowane jest przez właściwe mechanizmy tworzenia kopii zapasowych.

Istotny jest fakt, że platforma taka zawiera treści wymagające dużej pojemności dyskowej na przechowywanie. Zastosowanie standardowych mechanizmów tworzenia kopii zapasowych (ang. *backup*) jest problematyczne, ponieważ nie starcza czasu na wykonanie każdorazowo pełnego backupu. Stosowane jest więc połączenie mechanizmów backupów migawkowych z backupami przyrostowymi na dużych wolumenach danych charakteryzujących się mniejszą zmiennością. Polityka tworzenia kopii zapasowych określa również procedurę okresowego weryfikowania kopii zapasowych poprzez odtwarzanie systemów z tychże kopii (najczęściej „gdzieś na boku").

Drugie zagadnienie dotyczy zapewnienia odpowiedniej wydajności. W poprzednim rozdziale przytoczono mechanizmy realizacji tego problemu: skalowanie pionowe i poziome. Miarą oceny, czy system spełnia zakładane obciążenie, są przeprowadzane testy wydajnościowe. Platforma eNauczanie PG również podlegała takim testom. Przyjęto założenie 500 równoczesnych użytkowników i dla takiej ilości wyskalowano system.

Podczas przyjmowania zakładanego maksymalnego obciążenia warto mieć na uwadze, że obciążenie nie jest stałe w czasie – są okresy zarówno w czasie doby, jak i w semestrze akademickim, kiedy platforma będzie podlegała bardziej intensywnemu wykorzystaniu, i takie, kiedy mniejszemu. Dodatkowo założenie to nie może być stałe, ponieważ platforma ciągle się rozwija i tworzone są na niej coraz to nowe kursy, a więc jest coraz bardziej intensywnie wykorzystywana. Niezwykle istotne jest więc bieżące monitorowanie obciążenia i pracy platformy.

8. Wnioski końcowe

Platformy nauczania zdalnego ze względu na swoją wymaganą szeroką dostępność z praktycznie dowolnego urządzenia podpiętego do Internetu oraz na dużą złożoność rozwiązań technologicznych wystawione są na liczne zagrożenia. Z tego też powodu zapewnienie wymaganego poziomu bezpieczeństwa wymaga dużej ilości pracy. Istotnym jest, że proces ten jest ciągły – platforma powinna podlegać stałemu monitoringowi, okresowym audytom bezpieczeństwa, jak i testom wydajnościowym.

Nie bez znaczenia jest fakt, że dziedzina e-learningu podlega dynamicznemu rozwojowi, a więc i platformy je wspierające równie dynamicznie będą ewaluowały. Dotyczy to w szczególności rozwoju i łączenia różnych technologii świadczenia usług, jak i urządzeń, na których pracują użytkownicy (dynamiczny wzrost wykorzystania urządzeń mobilnych).

Stosowanie mechanizmów bezpieczeństwa zawsze ogranicza wygodę użytkownika na rzecz zapewnienia jak najwyższego poziomu bezpieczeństwa usługi. Należy jednak dążyć do właściwego kompromisu pomiędzy bezpieczeństwem a wygodą użytkowania, ponieważ system nawet bardzo bezpieczny będzie w małym stopniu wykorzystywany ze względu na jego niską użyteczność. Pomocna w tym zakresie jest analiza ryzyka zagrożeń i przyjęcie zakładanego poziomu akceptowalnego ryzyka. Konieczne jest też opracowanie planów działania na wypadek wystąpienia zagrożenia obejmujących naprawę jego skutków.

9. Bibliografia

1. Anisetti, M., Ardagna, C. A., Damiani, E., Saonara, F. (2013). A test-based security certification scheme for web services. ACM Transactions on the Web, 7, 2, 1–41.
2. Benantar, M. (2006). Access Control Systems. Security, Identity Management and Trust Models. Springer-Verlag US.
3. Cisco WebEx (2015). Pobrano 25 lutego 2015 z: http://www.webex.com
4. Clinch, J. (2009). ITIL v3 and information security. White Paper, 1–40.
5. Common Vulnerability Scoring System Version 2 Calculator. (2014). National Institute of Standards and Technology. Pobrano z: https://nvd.nist.gov/cvss.cfm?calculator&version=2
6. CVE Community,Common Vulnerabilities and Exposures. (2015). The MITRE Corporation. Pobrano 25 lutego 2015 z: http://cve.mitre.org
7. CWE Community,Common Weakness Enumeration. (2015). The MITRE Corporation, Pobrano 25 lutego 2015 z:

https://cwe.mitre.org

8. Damián-Reyes, P., Favela, J., Contreras-Castillo, J. (2009). Uncertainty Management in Context-Aware Applications: Increasing Usability and User Trust. Wireless Personal Communications, 56, 1,. 37–53.
9. De Capitani Di Vimercati, S., Foresti, S. Jajodia, S., Paraboschi, S., Psaila, G., Samarati, P. (2012). Integrating trust management and access control in data-intensive Web applications. ACM Transactions on the Web, 6, 2, 1–43.
10. Dimmock, N., Belokosztolszki, A., Eyers, D., Bacon, J., Moody, K. (2004). Using trust and risk in role-based access control policies. Proceedings of the ninth ACM symposium on Access control models and technologies – SACMAT '04, 156.
11. Disterer, G. (2013). ISO/IEC 27000, 27001 and 27002 for Information Security Management. Journal of Information Security, 4, 2, 92–100.
12. eNauczanie PG. (2015). Pobrano 25 lutego 2015 z: http://enauczanie.pg.gda.pl
13. Erl, T. (2007). SOA Principles of Service Design. SOA Systems Inc.
14. Goth, G. (2008). Single Sign-on and Social Networks. IEEE Distributed Systems Online, 9 (12), 1–1 .
15. Hansen, F., Oleshchuk, V. (2003). SRBAC: A spatial role-based access control model for mobile systems. Proceedings of the 7th Nordic Workshop on Secure IT Systems (NORDSEC'03), 129–141.
16. Hauser, J. R., Katz, G. M. (1998). Metrics: you are what you measure! European Management Journal, 4, 517-528.
17. Hinson, G. (2006). Seven myths about information security metrics. ISSA Journal, Pobrano 25 lutego 2015 z: http://www.noticebored.com/IsecT_paper_on_7_myths_of_infosec_metrics.pdf
18. Kirkpatrick, M. S., Bertino, E. (2010). Enforcing spatial constraints for mobile RBAC systems. SACMAT '10 Proceedings of the 15th ACM symposium on Access control models and technologies, 99–108.
19. Krawczyk, H, Lubomski, P. (2010). Generalized access control in hierarchical computer network. Zeszyty naukowe Wydziału Elektroniki, Telekomunikacji i Informatyki Politechniki Gdańskiej, 18, 217–222.
20. Krawczyk, H., Lubomski P. (2010).Pączkowanie – metoda rozwoju interoperacyjnych komponentów dla systemów rozproszonych. Inżynieria oprogramowania w procesach integracji systemów informatycznych, 8, 241–248.
21. Leavitt, N. (2011). Mobile Security: Finally a Serious Problem? Computer, 44, 6, 11–14.
22. Liu, Y., Hoang, D. B. (1994). OSI RPC model and protocol. Computer Communications, 17, 1, 53–66.
23. Lubomski, P., Żuchowski, I. (2014). Techniczne aspekty implementacji nowoczesnej platformy e-learningowej. Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej, 37, 41–44.
24. Lund, M. S., Solhaug, B., Stølen, K. (2010). Evolution in Relation to Risk and Trust Management. Computer, 43, 5, 49–55.
25. Mather, T., Kumaraswamy, S., Latif, S. (2009). Cloud Security and Privacy. An Enterprise Perspective on Risks and Compliance. O'Reilly.
26. Mell, P., Kent, K. A., Romanosky, S. (2007). The common vulnerability scoring system (CVSS) and its applicability to federal agency systems. US Department of Commerce, National Institute of Standards and Technology.
27. Mell, P., Scarfone, K., Romanosky, S. (2006). Common Vulnerability Scoring System, IEEE Security and Privacy Magazine, 4, 6, 85–89.
28. Mell, P., Scarfone, K., Romanosky, S. (2015). A Complete Guide to the Common Vulnerability Scoring System Version 2.0," FIRST.org, Inc. Pobrano 25 lutego 2015 z: http://www.first.org/cvss/cvss-guide
29. Microsoft Security Response Center Security Bulletin Severity Rating System (2002). Microsoft Developer Network, Pobrano 25 lutego 2015 z: http://msdn.microsoft.com/en-us/library/bb720758.aspx
30. Miller, C. (2011). Mobile Attacks and Defense. IEEE Security & Privacy Magazine, 9, 4, 68–70.
31. moodle. (2015). Pobrano 25 lutego 2015 z: https://moodle.org
32. National Vulnerability Database Version 2.2. (2015). National Institute of Standards and Technology. Pobrano 25 lutego 2015 z: https://nvd.nist.gov
33. NVD Common Vulnerability Scoring System Support v2. (2007). National Institute of Standards and Technology. Pobrano 25 lutego 2015 z: http://nvd.nist.gov/cvss.cfm
34. Open Sourced Vulnerability Database. (2015). Open Sourced Vulnerability Database (OSVDB). Pobrano 25 lutego 2015 z: http://osvdb.org
35. OWASP community. The Open Web Application Security Project (OWASP). (2015). Pobrano 25 lutego 2015 z: https://www.owasp.org
36. OWASP Top Ten Project. (2015). Pobrano 25 lutego 2015 z: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
37. Payne, S. C. (2006). A Guide to Security Metrics. SANS Security Essentials GSEC Practical Assignment, Pobrano 25 lutego 2015 z: https://www.sans.org/reading-room/whitepapers/auditing/guide-security-metrics-55.

38. Politechnika Gdańska, Moja PG. (2015). Pobrano 25 lutego 2015 z: https://moja.pg.gda.pl

39. SecurityFocus Vulnerability Database and BugTraq mail list. (2015). SecurityFocus. Pobrano 25 lutego 2015 z: http://www.securityfocus.com/vulnerabilities.

40. Software Engineering Institute, US-CERT Vulnerability Notes. (2015). Carnegie Mellon University. Pobrano 25 lutego 2015 z :https://www.kb.cert.org/vuls

41. The WASC Threat Classification v2.0. (2015) The Web Application Security Consortium (WASC). Pobrano 25 lutego 2015 z: http://projects.webappsec.org/w/page/13246978/Threat Classification

42. Thorne, K. (2003). Blended Learning: How to Integrate Online & Traditional Learning, Kogan Page.

43. US-CERT Vulnerability Metric (2015). National Institute of Standards and Technology. Pobrano 25 lutego 2015 z: www.kb.cert.org/vuls/html/fieldhelp#metric

44. Zaption (2015).Pobrano 25 lutego 2015 z: http://www.zaption.com

Security Challenges Of Modern E-Learning Platforms

Summary

Keywords: e-learning, security, interoperability, security audit, risk analysis, reliability, performance

Modern e-learning platforms are widely accessible at any time, from every place on the Earth. They use complex technology and are connected to many other systems. We can notice continuous growth and dynamic changes of such systems. On the other hand data processed by these systems are valuable and need to be protected. It is a big challenge to provide appropriate level of security and safety of such systems. The platforms being under consideration are mainly internet systems. Very often they combine many services developed in various technologies. They are exposed to threats similar to other internet systems. In the area of security the threats derive from errors in the implementation of authentication, authorization and session management. Oftentimes, weaknesses are the result of insufficient input validation. There is also a matter of ensuring service performance and availability at the intended level. To measure the level of security of the system, security audits are used. Each vulnerability detected by the audit should be analyzed in terms of the potential risk. Performance of the system is checked during performance tests. It is worth to notice, that it is constant work to measure, analyze and improve the level of system security.