Analiza oprogramowania Lab1

Raport wstępny z prac

Poznańskie Centrum

Superkomputerowo-Sieciowe

Dział Bezpieczeństwa ICT

Ul. Jana Pawła II 10, 61-139 Poznań

Poznań, grudzień 2017 r.

Spis treści

2 Streszczenie dla Zarządu

Przedmiotem zleconych testów była aplikacja Lab1 oparta o agenta PCL w wersji 2.25.6460.24770, zbierająca dane na temat zachowania użytkowników. Szczegółowy zakres monitoringu został określony jako:

Zbieranie czasu oraz informacji o użytkowniku w zakresie wykonywanych czynności:

- Włączenie komputera i wyłączenie komputera
- Zalogowanie, zablokowanie ekranu, wylogowanie, odblokowanie ekranu,
- Uruchomienie aplikacji i stron www,
- Tytuł uruchomionego okienka,
- Fakt kliknięcia klawiatury lub myszy bez wpisywanej treści.

Prowadzone prace polegały na analizie kodu, zarówno automatycznej jak i manualnej, oraz interakcji z aplikacją, prowadząc do obserwacji zachowania aplikacji oraz weryfikacji logowanych przez nią informacji. Dzięki uwzględnieniu w trakcie prac etapu analizy kodu, audytorzy mogli w pełni zweryfikować fakt braku istnienia ukrytej funkcjonalności, uruchamianej po zajściu określonego zdarzenia (np. przesłaniu specjalnego identyfikatora uruchamiającego tylną furtkę).

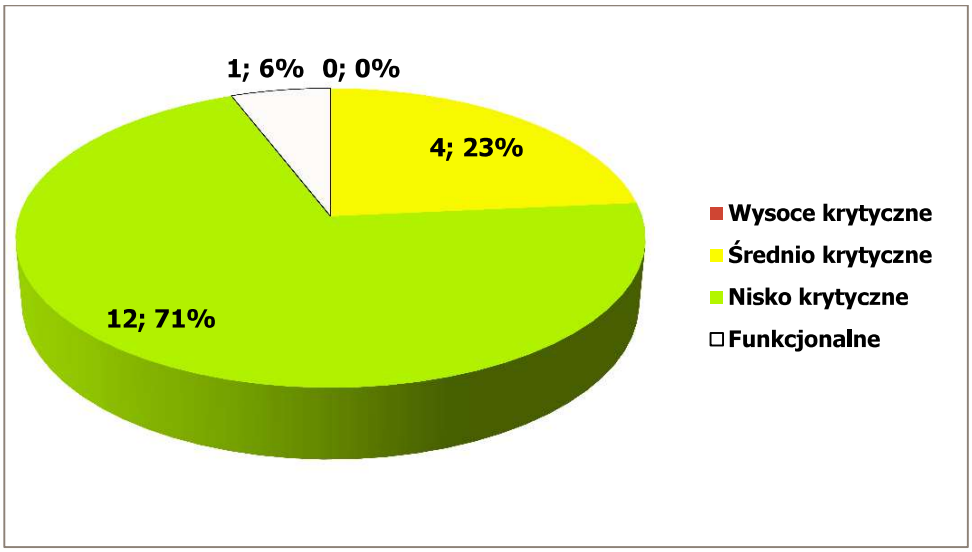
Testy kodu oraz interakcja z aplikacją potwierdziły, że **aplikacja nie wychodzi w monitorowaniu działań poza zdefiniowane wyżej założenia**.

Podczas analiz zauważono wprawdzie w kodzie miejsca, które mogły sugerować potencjalną możliwość odstępstw, jednak dokładniejsza ich analiza doprowadziła do potwierdzenia zgodności z definiowanym zakresem.

Opisane wyżej wątpliwości pojawiły się przy analizie modułu konfiguracji i dotyczyły funkcjonalności monitorowania wciśniętych przycisków dla klasy KeyLoggerPlugin z pakietu ActiveAppMonitor. Analiza wykazała tylko jedno odwołanie do danej funkcjonalności, z odpowiednią wartością parametru wejściowego, uniemożliwiającą dokładnego logowania wciśniętych przycisków.

Kolejna wątpliwość, to struktura opisana w błędzie o **ID 11**, gdzie analiza kodu wskazała brak odwołań do opisanego w tabelce konstruktora.

Ponadto audyt kodu oraz interakcja z aplikacją pozwoliły ujawnić 16 problemów bezpieczeństwa oraz jedną uwagę funkcjonalną. Należy zaznaczyć, że **nie znaleziono błędów wysoce krytycznych**.



Rysunek 1 – wykres przedstawiający procentowy rozkład znalezionych w oprogramowaniu błędów bezpieczeństwa

Spośród 16 błędów, zespół audytorski określił 4 z nich jako średnio krytyczne, natomiast pozostałe 12 jako nisko krytyczne.

W toku analizy zidentyfikowano krytyczne problemy, które w określonych warunkach mogłyby być uznane za krytyczne, jednak poziom zagrożenia z nimi związany został obniżony ze względu na zakładane ograniczenia w dostępie do problematycznej funkcjonalności oraz bardzo ograniczonym zakresie wpływu tych problemów na funkcjonalność. Do tego należy również podkreślić konieczność znajomości technik inżynierii odwrotnej w celu identyfikacji tych problemów – nie są to zaś kompetencje typowe dla użytkowników aplikacji, a nawet przeciętnie wykwalifikowanych agresorów sieciowych.

W porównaniu do innych aplikacji, które zespół audytorski miał okazję badać, kod aplikacji można ocenić jako **dobry plus**. Ocenę taką (a nie wyższą) determinuje fakt stosunkowo niewielkiej liczby błędów średnio krytycznych w odniesieniu do liczby błędów o najniższym stopniu krytyczności, a także bardzo duża liczba komentarzy w kodzie, sugerująca, że aplikacja nie jest jeszcze w pełni ukończona. Należy przy tym nadmienić, że niniejszy raport, a także ocena zespołu audytorskiego, dotyczą stanu zastanego na dzień otrzymania kodu do badań. W przypadku zmian w analizowanej aplikacji analizę należy powtórzyć.