STUDIA INFORMATICA 2012
Volume 33 Number 3A (107)

Krzysztof LASOTA
Ewa NIEWIADOMSKA-SZYNKIEWICZ
Adam KOZAKIEWICZ
Politechnika Warszawska, Instytut Automatyki i Informatyki Stosowanej[1]
Naukowa i Akademicka Sieć Komputerowa

ADAPTACJA ROZWIĄZAŃ HONEYPOT DLA SIECI CZUJNIKÓW

Streszczenie. Artykuł dotyczy zagadnień bezpieczeństwa sieci bezprzewodowych czujników, poczynając od przeglądu zagrożeń, po istniejące metody ochrony. Głównym tematem artykułu jest możliwość zastosowania idei honeypotów w sieciach WSN celem podniesienia poziomu bezpieczeństwa poprzez weryfikację poprawnego jej działania. Zaproponowano różne możliwości budowy systemu oraz wyodrębniono usługi, które mogą być chronione poprzez proponowane rozwiązanie.

Słowa kluczowe: honeypot, bezpieczeństwo teleinformatyczne, sieci sensorowe

ADAPTATION OF HONEYPOT SOLUTIONS FOR WSN

Summary. The article is devoted to security aspects of Wireless Sensor Networks. Starting with review and classification of existing threats and methods of improving security, the paper presents an idea of securing WSNs using honeypots for verification of its proper functioning. Several implementation possibilities are presented, as well as identification of the services which lend themselves to proposed mode of protection.

Keywords: honeypot, security, wireless sensor network

1. Wprowadzenie

Sieci sensorowe (WSN – *ang. Wireless Sensor Networks*) są szczególnym rodzajem sieci ad hoc, czyli samoorganizujących się sieci bezprzewodowych, których węzły są wyposażone

[1] Praca częściowo finansowana z projektu NCN nr NN514672940.

zarówno w nadajnik, jak i odbiornik sygnału radiowego. Ponadto, jak sama nazwa wskazuje, czujnik jest wyposażony w dodatkowy moduł wrażliwy na bodźce występujące w jego otoczeniu, np.: ruch, temperaturę, ciśnienie, wilgotność. W ostatnich latach sieci czujników są coraz powszechniej stosowane. Korzysta z nich wojsko, przemysł, rolnictwo, są wykorzystywane w opiece medycznej, stanowią podstawę inteligentnych domów. Mają być punktem wyjścia dla Internetu przyszłości, tzw. *Internet of Things*.

W niniejszym artykule dokonano przeglądu i przedstawiono klasyfikację zagrożeń sieci bezprzewodowych czujników (rozdział 2.) oraz zaprezentowano już istniejące rozwiązania, których zadaniem jest podwyższenie poziomu bezpieczeństwa sieci WSN (rozdział 3.). W rozdziale 4. przybliżono tematykę honeypotów, a następnie przedstawiono możliwość adaptacji tego rozwiązania na potrzeby sieci sensorowych (rozdział 5.).

2. Zagrożenia sieci sensorowych

W początkowym okresie prowadzenia badań nad sieciami czujników dość powszechnym podejściem było pomijanie problematyki bezpiecznego działania sieci. Zakładano, że w sieci występują jedynie zaufane węzły, które działają zgodnie z przyjętymi regułami i nie są podejmowane żadne próby wprowadzania błędnych informacji. Ze względu na bardzo szybki w ostatnich latach rozwój sieci bezprzewodowych sensorów, pominięcie aspektów bezpieczeństwa spowodowało, iż stały się one bardzo łatwym celem ataków.

2.1. Klasyfikacja ataków

W literaturze jest proponowanych wiele czynników, według których można dokonać klasyfikacji zagrożeń sieci sensorowych. Do najczęściej wymienianych kryteriów należą:

Aktywność atakującego: Ataki aktywne, w przypadku których atakujący bierze czynny udział (np.: sinkhole [1], wormhole [9], sybil [8], flooding [2]) oraz ataki pasywne, przeprowadzane przy biernej postawie agresora (np.: sniffing[2]).

Możliwości atakującego: Rozróżniamy ataki klasy sensor, gdy atakujący dysponuje porównywalnymi możliwościami w zakresie mocy obliczeniowej, ograniczeń zasilania, czy mocy sygnału co pojedynczy węzeł, oraz klasy laptop, w przypadku których atakujący dysponuje o wiele większymi możliwościami niż pojedynczy węzeł.

Położenie atakującego: Atakami wewnętrznymi nazywamy takie, w których węzeł atakujący należy do sieci, zewnętrznymi – pozostałe.

Atakowana warstwa: Ataki mogą odbywać się w różnych warstwach: fizycznej – atak na interfejs JTAG [10], łącza danych – atak na CSMA/CA [11], sieci – sinkhole, wormhole, sybil, transportowej/aplikacji – clone attack [3].

Cel ataku: Główne cele ataków to: uniemożliwienie komunikacji, podsłuch wiadomości, modyfikacja danych, generowanie fałszywych informacji.

3. Bezpieczeństwo sieci sensorowych

Wprowadzenie mechanizmów bezpieczeństwa w bezprzewodowych sieciach czujników było niezbędne. Większość zaprojektowanych standardów i zdefiniowanych w nich protokołów zakładało komunikację tylko między węzłami godnymi zaufania. Obecnie bezpieczeństwo sieci sensorowych to dynamicznie rozwijający się obszar badań – począwszy od bezpieczeństwa fizycznego, poprzez bezpieczeństwo energetyczne, bezpieczne protokoły (np. routingu), po bezpieczeństwo przekazywanych danych włącznie. Podobnie jak w klasycznych sieciach o stałej strukturze, bezpieczeństwo informacji zakłada: poufność – tylko osoby uprawnione posiadają dostęp do informacji, integralność – zagwarantowanie, że odebrane informacje opowiadają nadanym, aktualność – największe znaczenie przywiązuje się do informacji aktualnych, które nie powinny być ponownie wykorzystywane, oraz uwierzytelnienie – zagwarantowanie, że informacje pochodzą od nadawcy.

3.1. Warstwy bezpieczeństwa

Jednym z rozwiązań podwyższających poziom bezpieczeństwa sieci czujników bezprzewodowych jest wprowadzenie dodatkowej warstwy bezpieczeństwa w modelu działania sieci. Warstwa ta wykorzystuje algorytmy kryptograficzne i służy zabezpieczeniu przekazywanych pomiędzy węzłami informacji.

TinySec [4], prekursor, zapewnia integralność oraz poufność, ale nie wspiera aktualności ani uwierzytelnienia. Bardzo dużym problemem obniżającym poziom bezpieczeństwa sieci jest fakt, iż wszystkie węzły w sieci współdziedziczą ten sam klucz. *TinySec* był dedykowanym rozwiązaniem dla sieci czujników, gdzie pojedynczy węzeł ma bardzo ograniczone możliwości. Jego głównym atutem jest niska złożoność obliczeniowa, a tym samym niski narzut energetyczny.

ZigBee jest specyfikacją protokołów transmisji danych. Dodatkowo udostępnia funkcjonalność umożliwiającą korzystanie z kryptografii. Mechanizmy stosujące kryptografię symetryczną zapewniają wysoki poziom bezpieczeństwa sieci. Niestety, bezpieczeństwo jest

realizowane kosztem żywotności węzła – znaczna złożoność, a tym samym wysokie wymagania na moc obliczeniową podnoszą energochłonność rozwiązania.

MiniSec [6] w założeniu miał łączyć zalety obu wcześniej prezentowanych systemów – niskie narzuty *TinySec*-a oraz wysoki poziom bezpieczeństwa *ZigBee*. *MiniSec* wykorzystuje tryb szyfrowania blokowego OCB, który w jednym przebiegu zapewnia poufność oraz integralność danych. Dane są przetwarzane jednokrotnie, co obniża zużycie energii.

3.2. Systemy wykrywania włamań

Systemy wykrywania włamań są kolejnym rozwiązaniem podnoszącym poziom bezpieczeństwa sieci. Zadaniem systemów IDS (*ang. Intrusion Detection System*) jest zgłaszanie nietypowych sytuacji dotyczących sieci, identyfikowanych na podstawie zbieranych danych o jej zachowaniu. Dzięki wiedzy o pracy sieci jest możliwe wskazanie węzłów złośliwych lub rozpoznanie ich szkodliwego działania. Systemy IDS można klasyfikować na podstawie następujących kryteriów: sposób wykrywania (sygnatury, anomalie, zgodność w działaniu), zakres działania (cała sieć, grupa lub pojedynczy węzeł), tryb reakcji (aktywna, pasywna, zapobiegawcza). Parametry wykorzystywane przez systemy IDS do generowania alarmów to między innymi [13, 14]: współczynnik kolizji pakietów, czas oczekiwania na otrzymanie pakietu, liczba sąsiadów, zmiany metryki ścieżki routingowej, szybkość zużycia energii, współczynnik gubienia pakietów, procent integralnych wiadomości, współczynnik wiadomości odrzuconych przez danego sąsiada.

3.3. Watchdog

Mechanizm *watchdog*, czyli tzw. pies stróżujący, polega na obserwacji i klasyfikacji, czy system działa poprawnie. Pierwotnie *watchdog* był instalowany w systemach działających bez interakcji z człowiekiem, których nieprzerwane funkcjonowanie było bardzo ważne, a nieprzewidziane działanie mogło być niebezpieczne. Watchdog WSN [12] wykorzystuje fakt, iż sieć sensorowa korzysta ze współdzielonego medium.

W sieci składającej się z trzech węzłów uproszczone działanie mechanizmu można opisać w czterech krokach:

- krok 1: węzeł A chce wysłać wiadomość do C, przy czym C jest poza zasięgiem A,
- krok 2: węzeł A wysyła wiadomość do B, aby ten przekazał ją węzłowi C,
- krok 3: węzeł A, po zakończeniu transmisji do B, nadal obserwuje kanał, sprawdzając, czy węzeł B wysłał wiadomość do C,
- krok 4: jeśli węzeł B wysłał wiadomość do C, A zakłada, że system działa poprawnie, jeśli nie, wnioskuje, że węzeł B działa fałszywie.

4. Honeypot

Honeypot (dosłownie garniec miodu) jest pułapką, mającą na celu wykrycie prób nieautoryzowanego użycia systemu lub pozyskania danych. Najczęściej, pułapka jest realizowana jako niezabezpieczony lub bardzo słabo zabezpieczony system, bądź usługa. Celem jest skuszenie potencjalnego agresora, aby zaatakował właśnie pułapkę, a nie inny zasób. Honeypoty możemy podzielić ze względu na sposób wyszukiwania zagrożeń:

Serwerowy: Pasywnie wyczekuje atakującego, udostępniając niezabezpieczone usługi, takie jak telnet, czy www. Przykładem takiego rozwiązania jest m.in. Dionaea.

Klienckie: Aktywnie poszukują zagrożeń. Wysyłane są żądania i odbierane odpowiedzi, np.: honeyC symuluje działanie przeglądarki internetowej.

Inną klasyfikację można przedstawić w zależności od działania samej pułapki:

Niskointeraktywne: Symulują działanie systemu/usługi. Ocena, czy nastąpił atak, jest formułowana bezpośrednio na podstawie danych otrzymanych tylko podczas komunikacji, np.: w przypadku symulacji przeglądarki internetowej przez niskointeraktywnego honeypota badana jest zawartość strony. Poszukuje się m.in. podejrzanych ciągów znaków bądź przekierowań do innych stron. Przykładem takiego rozwiązania jest m.in. Monkey-Spider.

Wysokointeraktywne: Zakładają emulację w pełni funkcjonalnych systemów bądź usług. Decyzja o stwierdzeniu zagrożenia jest podejmowana na podstawie zdarzeń zachodzących w systemie. Obserwowany jest rejestr systemu, stan pamięci, operacje dyskowe itp. Przykładem takiej pułapki jest m.in. SHELIA.

Hybrydowe: łączą w sobie funkcjonalność nisko- i wysokointeraktywnych honeypotów. Jak na razie jedynym znanym autorom przedstawicielem tej grupy jest HoneySpider Network.

5. Honeypot WSN

Sieci sensorowe charakteryzują się ograniczonymi zasobami. Biorąc pod uwagę możliwości pojedynczych sensorów, nie wszystkie rozwiązania stosowane w sieciach komputerowych o stałej strukturze mogą być zaadaptowane do sieci czujników. W związku z tym intensywnie poszukuje się alternatywnych technologii podnoszących poziom ochrony sieci WSN. W rozdziale 3 przedstawiono stosowane i prezentowane w literaturze mechanizmy. W niniejszym proponowane jest rozwiązanie, stosujące ideę honeypotów.

Zakłada się, że zadaniem honeypota WSN jest weryfikacja poprawnej pracy sieci poprzez pewne udostępniane usługi. Zależnie od trybu pracy – serwerowy lub kliencki – honeypot sprawdzałby, czy system lub pojedynczy węzeł zachowuje się zgodnie z oczekiwaniami.

5.1. Usługi

Adaptację honeypotów na potrzeby bezprzewodowych sieci sensorowych rozpoczęto od wyodrębnienia usług możliwych do wykorzystania przez klienckie bądź serwerowe honeypoty. Najciekawsze z użytkowego punktu widzenia możliwości dają następujące usługi:

Synchronizacja czasu w węzłach: Czas jest wykorzystywany do synchronizacji pracy sieci. W wielu systemach znacznik czasu jest wymaganym elementem przekazywanych wiadomości. Znacznik czasu jest wykorzystywany m.in. przez systemy kryptograficzne, systemy dystrybucji kluczy (np.: LEAP[5]) oraz zapewnia aktualność danych.

Lokalizacja węzła/topologia: Wykorzystywana przede wszystkim przez algorytmy routingu optymalizujące koszt ścieżek.

Przekazywanie komunikatów: Podstawowa usługa każdej sieci ma na celu przekazanie komunikatu od odbiorcy do adresata.

Wykonanie pomiaru mierzonej zmiennej: ruch, temperatura itp.

Aktualizacja oprogramowania: W rozbudowanych sieciach sensorowych istnieje możliwość zdalnej aktualizacji oprogramowania na każdym z czujników. Ułatwia to administrację systemem.

Prawidłowe działanie przedstawionych usług jest wymagane do poprawnej pracy sieci. Honeypoty umieszczone w sieci mogą być wykorzystywane do weryfikacji poprawności działania konkretnych usług. Pierwsze cztery usługi mogą być weryfikowane za pomocą klienckich honeypotów, natomiast usługa aktualizacji oprogramowania wymaga użycia honeypota serwerowego.

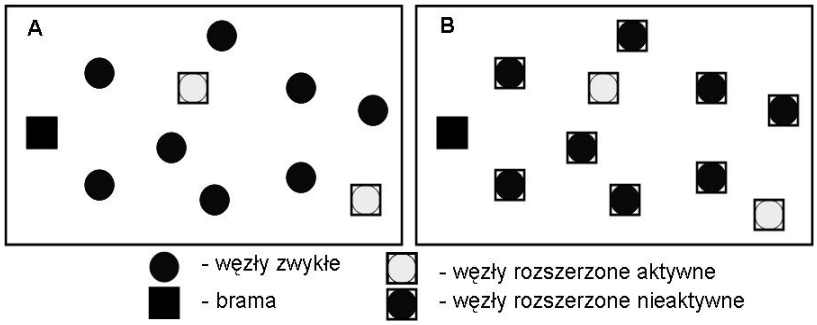
5.2. Architektura

Opracowano wstępną wersję architektury systemu. Wykorzystując doświadczenia przy tworzeniu systemów wykrywania włamań przeznaczonych dla sieci WSN [7, 15], zaproponowano dwa rozwiązania. Pierwsze rozwiązanie, przedstawione na rys. 1a, zakłada istnienie w sieci dwóch rodzajów węzłów. Są to tzw. węzły zwykłe, posiadające podstawową funkcjonalność czujnika oraz węzły rozszerzone wyposażone w dodatkową funkcjonalność honeypota. Ograniczeniem takiego rozwiązania jest możliwość weryfikacji usług jedynie tych czujników, które mogą nawiązać dwustronną komunikację z węzłem o rozszerzonej funkcjonalności.

Druga zaproponowana architektura systemu (rys. 1b) zakłada, iż w sieci znajdują się wyłącznie węzły o rozszerzonej funkcjonalności. Przyjmuje się, że w celu oszczędzania baterii, tylko niektóre węzły posiadają włączoną funkcjonalność honeypota. Włączanie i wyłączanie dodatkowej funkcjonalności może być zorganizowane na kilka sposobów. Prezentujemy dwa z nich:

Wykorzystanie tokena: Przyjmuje się, że po sieci krążą specjalne wiadomości, tzw. tokeny. Węzeł, który posiada token, posiada również włączoną dodatkową funkcjonalność.

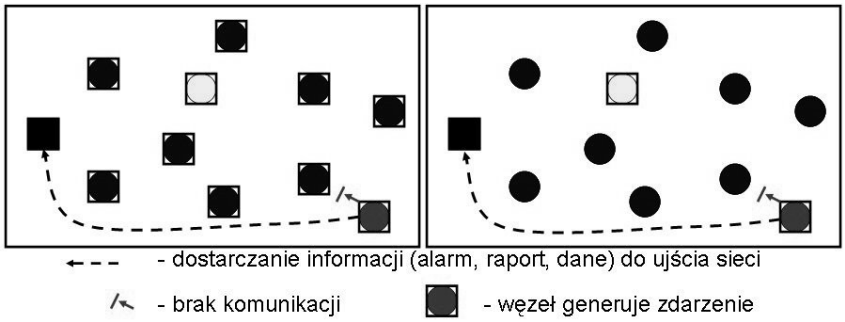
Włączanie losowe: Dodatkowe funkcje w węzłach są włączane niezależnie. Każdy węzeł sam losuje, kiedy i jak jest wykorzystywana funkcjonalność honeypota.



Rys. 1. Architektura: A – dwa rodzaje węzłów, B – jeden rodzaj
Fig. 1. Architecture: A – two types of nodes, B – one type

Niezależnie od przyjętego rozwiązania specyfikacji wymaga protokół komunikacyjny używany do wymiany informacji pomiędzy węzłami, jak również ujściem sieci. Najważniejsze typy wiadomości to:

- ALARM – wysyłana z węzła do ujścia sieci w przypadku wykrycia nieprawidłowości,
- ALARM_ALL – wysyłana z węzła do sąsiadów w przypadku wykrycia nieprawidłowości, może również wpływać na reputację nadawcy,
- ALARM_CON – potwierdzenie wykrycia nieprawidłowości, przekazywane od bramy do pozostałych węzłów w sieci, prócz uznanych za nieuczciwe,
- RAPORT – okresowe dane (np.: ilość incydentów bezpieczeństwa – otrzymanych, wykrytych, potwierdzonych) przekazywane do ujścia sieci,
- CHECK – wymuszenie sprawdzenia konkretnej usługi przez konkretny węzeł, u wybranego lub wszystkich sąsiadów. Wiadomość obsługiwana wyłącznie przez węzły posiadające funkcjonalność klienckiego honeypota.



Rys. 2. Usługa krytyczna
Fig. 2. Critical service

5.3. Ograniczenia

Zastosowanie rozwiązań honeypot ma niestety swoje ograniczenia. Zarówno w ich przypadku, jak również w rozproszonych systemach wykrywania włamań lub innych systemach podnoszących poziom bezpieczeństwa sieci WSN, krytyczną usługą jest możliwość propagacji danych (rys. 2.). W przypadku kiedy przekazanie informacji o wygenerowanych alarmach, okresowych raportów itp. z węzła do ujścia sieci jest niemożliwe, można uznać, że system został skompromitowany, nie działa poprawnie, a więc traci swoje możliwości wykrywania zagrożeń.

6. Podsumowanie i dalsze prace

W artykule zaprezentowano możliwość wykorzystania honeypotów w sieciach bezprzewodowych czujników. W zaproponowanym rozwiązaniu funkcjonalność honeypotów jest realizowana poprzez weryfikację poprawnego działania wyodrębnionych usług, co przyczynia się do podniesienia poziomu bezpieczeństwa pojedynczego czujnika, jak i całej sieci. Zastosowanie znajdą zarówno honeypoty serwerowe (aktualizacja oprogramowania), jak również klienckie (synchronizacja czasu w węzłach, usługa lokalizacji węzła/topologia, przekazywanie komunikatów, pomiar wartości), niezależnie od wybranej architektury systemu. Biorąc pod uwagę ograniczone zasoby czujników, uwaga koncentrowała się na niskointeraktywnym honeypocie.

Celem dalszych prac jest zakończenie implementacji prototypowego systemu, składającego się z trzech elementów. Kliencki oraz serwerowy honeypot jest wykonywany przy użyciu frameworku TinyOS. Aplikacja zarządzająca w formie konsolowego narzędzia jest pisana w C++. Wytworzone oprogramowanie zostanie poddane testom w środowisku laboratoryjnym oraz eksperymentalnej ocenie w sieci złożonej z ponad stu czujników (x55 CM3000 oraz x50 CM5000 firmy Advanticsys).

BIBLIOGRAFIA

1. Singh S. K., Singh M. P., Singh D. K.: A Survey on Network Security and Attack Defense Mechanism For Wireless Sensor Networks. International Journal of Computer Trends and Technology – May to June Issue 2011, ISSN: 2231-2803.
2. Kalita H.K., Kar A.: Wireless Sensor Networks Security Analysis, International Journal of Next-Generation Networks (IJNGN), vol. 1, no. 1, Dec. 2009, s. 1÷9.

3. Bojkovic Z. S., Bakmaz B. M., Bakmaz M. R.: Security Issues in Wireless Sensor Networks. NAUN Int. Journal of Communications, 2008, vol.2, no.1, s. 106÷115.
4. Karlof C., Sastry N., Wagner D.: TinySec: A Link Layer security Architecture for Wireless Sensor Networks, ACM SenSys 2004, Nov. 3-5, 2004, s. 162÷175.
5. Zhu S., Setia S., Jajodia S.: Leap: efficient security mechanisms for large scale distributed sensor networks. In CCS '03: Proceedings of the 10th ACM conference on Computer and communications security, New York, USA, 2003, s. 62÷72.
6. Luk M., Mezzour G., Perrig A., Gligor V.: MiniSec: a secure sensor network communication architecture. Proceedings of the 6th International Conf. on Information Processing in Sensor networks, April, 2007, Cambridge, Massachusetts, USA.
7. Stępień A.: Wykrywanie anomalii w pracy sieci sensorowej. Praca magisterska, Politechnika Warszawska, WEITI, marzec 2011.
8. Newsome J., Shi E., Song D., Perrig A.: The Sybil attack in sensor networks analysis & defenses, International Symposium on Information Processing in Sensor Network, vol. 1 2004, s. 259÷268.
9. Hu Y. C., Perrig A., Johnson D. B.: Packed Leashes: a defense against wormhole attacks in wireless networks, IEEE INFOCOM, vol. 1, 2003, s. 1976÷1986.
10. Skorobogatov S. P.: Semi-invasive attacks – a new approach to hardware security analysis. Tech. Report UCAM-CL-TR-630, University of Cambridge, Comp. Lab., April 2005.
11. Yang Xiao: Security in sensor network. Auerbach Publications, 2007.
12. Hu F., Ziobro J., Tillett J., Sharma N. K.: Secure wireless sensor networks: Problems and solutions. Journal of Systemics, Cybernetics and Informatics, vol.1, no.4, s. 90÷100, 2003.
13. Demirkol I., Alagoz F., Delic H., Ersoy: Wireless Sensor Networks for Intrusion Detection: Packet Traffic Modeling, Comm. Letters, no.1, vol. 10, s. 22÷24, IEEE, 2006.
14. Wong M. C., Xiao Y., Su X.: Security Issues in Ad Hoc Networks. Xiao Y. (red.): Security in Sensor Networks. Auerbach Publications, 2007, s. 215÷236.
15. Huo G., Wang X.: DIDS: A Dynamic Model of Intrusion Detection System in Wireless Sensor Networks. Proceedings of the 2008 IEEE International Conference on Information and Automation, 2008 Zhangjiajie, Chiny, s. 374÷378.
16. Deng H., Li W., Agrawal D. P.: Routing Security in wireless ad hoc networks. IEEE Communication Magazine, vol. 40 no. 10, 2002, s. 70÷75.
17. Mishra A., Nadkarni K., Patcha A.: Intrusion detection in wireless ad hoc networks. IEEE Wireless Communications, vol. 11 no 1, 2004, s. 48÷60.
1. Karlof C., Wagner D.: Secure routing in wireless sensor networks attacks and countermeasures, IEEE International Workshop on Sensor Network Protocols and Applications, vol. 1, 2003, s. 113÷127.

18. Hu Y. C., Perrig A., Johnson D. B.: Rushing attacks and defense in wireless ad hoc network routing protocols, ACM Workshop on Wireless Security, vol. 1, 2003, s. 30÷40.

Wpłynęło do Redakcji 14 marca 2012 r.

Abstract

The article is devoted to security aspects of Wireless Sensor Networks. Starting with review and classification of existing threats (Chapter 2), it describes methods of improving security using valued existing state-of-the-art solutions (Chapter 3). Chapter 4 outlines the topic of honeypots and chapter 5 presents an idea of using honeypots in WSNs.

Research on honeypot-based methods of improving WSN security proved that honeypots can indeed be adapted for this type of networks. The solution proposes using honeypots for detecting attacks against a preselected set of services. Both types of honeypots, server-side (e.g. monitoring of firmware updates) and client-side (e.g. monitoring of time synchronization process, node location services, message transfer, sensor operation, etc.), are useful regardless of the architecture of the monitored system (Fig 1.). Further research of system architecture is to be conducted to reduce limitations described in the article (e.g. Fig 2.). Implementation of a proof-of-concept solution is limited to a low-interaction honeypot, due to limited resources of sensor nodes. It will be tested in lab environment and subjected to experimental evaluation.

Adresy

Krzysztof LASOTA: Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych, ul. Nowowiejska 15/19, 00-665 Warszawa, Polska, klasota@mion.elka.pw.edu.pl; Naukowa i Akademicka Sieć Komputerowa, ul. Wąwozowa 18, 02-796 Warszawa, Polska, krzysztof.lasota@nask.pl

Ewa NIEWIADOMSKA-SZYNKIEWICZ: Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych, ul. Nowowiejska 15/19, 00-665 Warszawa, Polska, ens@ia.pw.edu.pl; Naukowa i Akademicka Sieć Komputerowa, ul. Wąwozowa 18, 02-796 Warszawa, Polska, ewan@nask.pl

Adam KOZAKIEWICZ: Politechnika Warszawska, Wydział Elektroniki i Technik Informacyjnych, ul. Nowowiejska 15/19, 00-665 Warszawa, Polska, akozakie@elka.pw.edu.pl; Naukowa i Akademicka Sieć Komputerowa, ul. Wąwozowa 18, 02-796 Warszawa, Polska, adam.kozakiewicz@nask.pl