ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 2014
Seria: ORGANIZACJA I ZARZĄDZANIE z. 74 Nr kol. 1921

Wojciech TYLMAN
Katedra Mikroelektroniki i Technik Informatycznych
Politechnika Łódzka

OCHRONA PRZEMYSŁOWYCH SYSTEMÓW STEROWANIA PRZEZ ANALIZĘ RUCHU SIECIOWEGO

Streszczenie. Przedstawiona jest koncepcja wysoce zautomatyzowanego rozwiązania pozwalającego na wykrywanie w przemysłowym ruchu sieciowym sytuacji odbiegających od stanu normalnego (anomalii). Omówione są zastosowania klasycznych sieci bayesowskich i sieci Multi-Entity Bayesian Networks (MEBN) wraz z dyskusją ich stosowalności w praktyce. Prace ilustrują również możliwość wykorzystania istniejącego oprogramowania (na przykładzie systemu Snort) oraz kwestie wymaganych modyfikacji związanych z pracą w sieciach nie-IP.

Słowa kluczowe: przemysłowe systemy sterowania, sieci przemysłowe, wykrywanie anomalii, sieci bayesowskie, sieci MEBN.

PROTECTION OF INDUSTRIAL CONTROL SYSTEMS THROUGH ANALYSIS OF NETWORK TRAFFIC

Summary. The paper presents a concept of a highly automated solution allowing detection, in industrial network traffic, of situations differing from the normal state (anomalies). It describes the use of classical Bayesian networks and Multi-Entity Bayesian Networks (MEBN), together with a discussion of their applicability in practice. The work also illustrates the possibility of using existing software (taking Snort system as an example) and the required modifications related to the support for non-IP networks.

Keywords: industrial control systems, industrial networks, anomaly detection, Bayesian networks, MEBN networks.

1. Wstęp

Dzisiejsze przemysłowe systemy sterowania masowo wykorzystują komunikację sieciową. Połączenia sieciowe są stosowane zarówno pomiędzy sterownikami przemysłowymi (PLC, ang. Programmable Logic Controller), stanowiącymi warstwę bezpośrednio odpowiedzialną za sterowanie elementami procesu produkcyjnego, jak też pomiędzy sterownikami PLC i komputerami sterowania nadrzędnego – stanowią tym samym istotny element systemów SCADA (ang. Supervisory Control And Data Acquisition). Wykorzystanie połączeń sieciowych pozwala na wygodne tworzenie złożonych, rozproszonych systemów sterowania, stwarza możliwość zdalnego nadzoru nad procesem produkcji, wygodnego rekonfigurowania i diagnostyki.

Taka sytuacja oznacza jednak, że dane kluczowe dla procesu produkcji są przesyłane zbiorczo przez niewielką liczbę kanałów komunikacyjnych, a także, że możliwy jest zdalny dostęp do niemalże wszystkich elementów sterujących produkcją. Stwarza to zagrożenie związane z możliwością nieuprawnionej ingerencji w działanie takiego systemu. Co gorsza, ingerencja ta może nastąpić bez fizycznej obecności w pobliżu zakłócanego urządzenia.

Należy przy tym zauważyć, że tematyka wykrywania takich zdarzeń jest bardzo rzadko poruszana w literaturze przedmiotu. Oczywiście projektanci systemów zdają sobie sprawę z ryzyka związanego z nieuprawnionym dostępem do sieci, jednak ochrona sieciowych systemów sterowania w praktyce sprowadza się do stosowania ustawień konfiguracyjnych minimalizujących ryzyko takiego zdarzenia. Kwestia wykrywania sytuacji, kiedy osoba postronna uzyskała jednak dostęp do sieci i elementów systemu sterowania, jest marginalizowana. Stan ten można uznać za zaskakujący: w przypadku sieci Internet prace nad wykrywaniem takich sytuacji są bowiem bardzo liczne.

Jednocześnie coraz częściej pojawiają się doniesienia o zakończonych sukcesem atakach na przemysłowe systemy sterowania; jednym z najbardziej spektakularnych przykładów jest zniszczenie części urządzeń zakładu wzbogacania uranu w Natzan (Iran) przez specjalnie zaprojektowanego robaka Stuxnet (2010). Wydarzenie to zasługuje na bardziej szczegółowe omówienie, pokazuje bowiem, jakie możliwości otwierają przed atakującym skomputeryzowane sieciowe systemy sterowania. Robak Stuxnet został tak zaprojektowany, aby infekować określony typ sterownika Siemens S7. Pierwszym celem ataku były komputery PC należące do systemu SCADA; były one infekowane przez dyski przenośne i komunikację sieciową typu *peer-to-peer*. W przypadku podłączenia sterownika do zainfekowanego komputera PC za pomocą przewodu transmisji danych następowało przeprogramowanie sterownika, tak aby zmienić charakterystykę pracy sterowników wirówek, które to sterowniki były podłączone do sieci przemysłowej Profibus. Zmieniona charakterystyka pracy powodowała przeciążenie elementów mechanicznych wirówki, która po pewnym czasie ulegała *fizycznemu zniszczeniu*.

Powszechne stosowanie komunikacji sieciowej otwiera jednak również nowe możliwości w kwestii zapewnienia bezpieczeństwa, umożliwia bowiem monitorowanie działania systemu przez monitorowanie ruchu sieciowego, przy czym użyteczność takiego monitorowania wykracza poza kwestię wykrywania wyżej wspomnianych ingerencji: w ruchu sieciowym można bowiem dostrzec oznaki nieprawidłowości działania systemu, spowodowanych innymi czynnikami, np. niepoprawnym działaniem czujników.

W tej pracy zaprezentowano strategię monitorowania ruchu sieciowego w sieciach przemysłowych, mającą na celu wykrywanie sytuacji odbiegających od stanu normalnego (anomalii) z wykorzystaniem sieci bayesowskich. Prezentowane podejście wykorzystuje przy tym elementy oprogramowania stosowanego do wykrywania ataków w sieci Internet – ma to przyspieszyć prace i dać dostęp do bogatego zestawu oprogramowania towarzyszącego (związanego np. z wizualizacją czy też analizą wyników pracy systemu detekcji).

2. Dotychczasowy stań badań związanych z wykrywaniem ataków w przemysłowych systemach sterowania

W literaturze przedmiotu można znaleźć pojedyncze publikacje dotyczące systemów wykrywania ataków w przemysłowych systemach sterowania. W pracach tych autorzy sugerują możliwość wykorzystania wybranych metod „soft computing", np. logiki rozmytej [4] bądź sieci neuronowych [6]. Innym podejściem jest koncepcja zaprezentowana w [2], gdzie autorzy wykorzystują analizę n-gramów do określenia typowego zachowania sieci energetycznych. Analiza n-gramów zostanie omówiona w dalszym ciągu niniejszego artykułu, stanowi bowiem jeden z elementów zaproponowanego systemu. Istotność komunikacji sieciowej jest podnoszona tylko w niektórych publikacjach [6].

Jeśli chodzi o praktyczne wykorzystanie sieci bayesowskich w omawianym zagadnieniu zgodnie z najlepszą wiedzą autora brak jest doniesień literaturowych. Należy natomiast wspomnieć o dwu projektach, w których możliwość zastosowania podejścia bayesowskiego, w tym sieci bayesowskich, była sygnalizowana. Pierwszy z tych projektów był finansowany przez Departament Bezpieczeństwa Krajowego Stanów Zjednoczonych i dotyczył wykrywania ataków w systemach SCADA [3, 9, 10]. Przedstawiał on zalety prowadzenia detekcji na podstawie modelu chronionego systemu; taka koncepcja została również wykorzystana w pracach autora. Drugi projekt, nazwany Viking, dotyczył przede wszystkim bezpieczeństwa sieci energetycznych i był finansowany w ramach 7. Programu Ramowego Unii Europejskiej [14, 8].

W ramach dyskusji należy jeszcze wspomnieć o sygnalizowanym przez niektórych autorów [4] związku pomiędzy wykrywaniem ataków a nieprawidłowościami w działaniu systemów przemysłowych spowodowanymi przez inne przyczyny (np. awarie). Jest to o tyle

istotne, że prace dotyczące wykrywania tego typu sytuacji są już prowadzone od dłuższego czasu [1]. Również w przypadku prac autora na temat monitorowania ruchu sieciowego należy podkreślić możliwość wykrywania nie tylko celowych ataków, lecz także awarii: również one mogą bowiem znaleźć odzwierciedlenie w obserwowanym ruchu sieciowym.

3. Sieci bayesowskie

Sieci bayesowskie [7] są skierowanymi grafami acyklicznymi (DAG, ang. Directed Acyclic Graph), opisują one jakościowe związki probabilistyczne pomiędzy zmiennymi w postaci graficznej. Węzły grafu odpowiadają zmiennym, a krawędzie wskazują na zależności między nimi. Przez brak połączenia między węzłami można wskazać na niezależność zmiennych, co z kolei umożliwia zakodowanie rozkładu prawdopodobieństwa zmiennych w bardzo kompaktowy sposób. Oprócz grafu do utworzenia kompletnego opisu konieczne jest podanie wartości liczbowych: prawdopodobieństw warunkowych i bezwarunkowych.

Sieć bayesowską można utworzyć zarówno ręcznie, jak i za pomocą uczenia maszynowego z danych. Wygoda ręcznego tworzenia sieci, wynikająca ze zrozumiałości opisu graficznego, stanowi ważny atut sieci bayesowskich – pozwala bowiem na wygodne zakodowanie wiedzy eksperckiej.

Po utworzeniu sieci możliwe jest wykonanie wnioskowania przez ustawienie stanu obserwowanych zmiennych i zastosowanie jednego z licznych algorytmów, które pozwalają na obliczenie prawdopodobieństwa pozostałych zmiennych.

Sieci bayesowskie w swej klasycznej postaci nie są pozbawione wad. W literaturze przedmiotu podnosi się takie kwestie, jak brak reprezentacji obiektów (co powoduje niemożność opisu sytuacji ze zmienną liczbą parametrów), wsparcia dla opisu stanu zmiennej za pomocą wyrażeń logicznych, wsparcia dla zależności czasowych. Jednym z najnowszych rozszerzeń mających na celu likwidację tych niedociągnięć są sieci MEBN (ang. Multi Entity Bayesian Networks) [5]. Ich praktyczna przydatność nie została dotychczas oceniona, choć dostępne jest już oprogramowanie implementujące odpowiednie algorytmy i pozwalające na konstruowanie sieci MEBN (jednak bez wsparcia dla uczenia maszynowego) oraz wnioskowanie przy ich wykorzystaniu [13].

4. System wykrywania anomalii przy wykorzystaniu monitorowania ruchu sieciowego

4.1. Wykorzystanie istniejącego oprogramowania a problem sieci nie-IP

Jak już wspomniano, z praktycznego punktu widzenia związanego z możliwością wdrożenia proponowanej strategii ochrony systemów przemysłowych istotne jest wykorzystanie istniejącego oprogramowania, pisanego pod kątem wykrywania ataków w sieci Internet. Na rynku dostępnych jest wiele produktów; szczególną uwagę należy zwrócić na rozwiązania z otwartym kodem z uwagi na łatwość dostosowania ich do nowych zastosowań. Wśród takich rozwiązań na uwagę zasługuje system Snort [12], który można uznać za standard w sieciowej detekcji ataków. Snort nie może być jednak bezpośrednio użyty jako podstawa proponowanego systemu z powodu braku wsparcia dla przetwarzania danych pochodzących z sieci nie-IP.

Problem ten wynika z ujednolicenia protokołów wykorzystywanych w Internecie (gdzie zawsze obecny jest protokół IP) oraz różnorodności sieci przemysłowych (gdzie protokół IP jest wykorzystywany tylko w niektórych). Tym samym pierwszym etapem prac było dostosowanie systemu Snort do pracy w sieciach nie-IP. Jako przykład wybrano protokół Modbus RTU działający na podstawie interfejsu RS-485.

Prace zostały ułatwione przez wprowadzenie w wersji 2.9 systemu Snort modułów DAQ (ang. Data Acquisition Library), czyli wymienialnych elementów odpowiedzialnych za przechwytywanie danych z sieci. Korzystając z tego podejścia, autor opracował moduł DAQ dostosowany do charakterystyki interfejsu RS-485 i protokołu Modbus RTU. Jednakże samo przechwycenie danych nie stanowi rozwiązania problemu: cała ścieżka przetwarzania w programie Snort jest dostosowana do protokołów stosu TCP-IP. Aczkolwiek możliwe są rozbudowanie i modyfikacja tej ścieżki, wymagałoby to dużego nakładu pracy i oznaczało potencjalny brak kompatybilności z kolejnymi wersjami programu Snort (w związku z koniecznością modyfikacji kodu źródłowego). Prostota protokołu Modbus RTU pozwoliła jednak na inne rozwiązanie: dla każdej przechwyconej ramki protokołu Modbus RTU generowana jest sztuczna ramka protokołu UDP, wspieranego przez system Snort. Następnie treść przechwyconej ramki jest wstawiana jako treść ramki UDP. Wymogi odnośnie do długości ramki protokołu Modbus RTU powodują, że taka enkapsulacja jest zawsze możliwa.

4.2. Modelowanie dopuszczalnych schematów komunikacji

Podejście przedstawione w tym podpunkcie opiera się na obserwacji dotyczącej przebiegu komunikacji w sieciach przemysłowych. W przeciwieństwie do sieci Internet, w której dany komputer zwykle łączy się z wieloma innymi komputerami (których nie da się traktować jako zamkniętą grupę), a nawiązywane połączenia mogą się znacznie od siebie różnić

(wykorzystywać różne protokoły, mieć różne czasy trwania, ilości przesyłanych danych, liczbę pakietów itp.), w sieciach przemysłowych schemat komunikacji danego elementu (np. sterownika PLC) jest bardzo dobrze określony: łączy się on zawsze z komputerami z niewielkiej, zamkniętej grupy, wykorzystuje zawsze ten sam protokół komunikacyjny. Podobne – albo wręcz identyczne – są też wielkości charakteryzujące kolejne transmisje. Sytuacje, w których schemat komunikacji odbiega od typowego, mogą świadczyć o ataku bądź niepoprawnej pracy elementu.

W związku z tym autor wybrał następujące parametry, charakteryzujące każdorazową wymianę danych pomiędzy dwoma urządzeniami podłączonymi do sieci:

- rozmiar w bajtach,
- rozmiar w pakietach,
- średni rozmiar pakietu,
- odchylenie standardowe rozmiaru pakietu,
- czas trwania.

W fazie uczenia zbierane są parametry obserwowanych wymian danych, są one następnie wykorzystywane w procesie tworzenia sieci bayesowskiej (zob. podpunkt 4.4).

4.3. Analiza treści komunikacji

Również pod względem przesyłanej treści komunikacja w sieciach przemysłowych jest mało zróżnicowana. Widać to np. gdy mamy do czynienia z przesyłem danych z czujników: podczas normalnej pracy dane takie będą się mieściły w ściśle określonych przedziałach, zdeterminowanych przez wymogi poprawnej pracy urządzeń. Odstępstwa od normy mogą świadczyć o celowej próbie zakłócenia pracy urządzeń (jak np. w przypadku robaka Stuxnet), ale również o awarii obiektu sterowanego lub czujnika.

Wartości typowe mogą być określone przez projektanta systemu wykrywania anomalii, jednak jest to sprzeczne z założeniem pracy w pełni automatycznej. Podobnie założenie to nie pozwala na wprowadzenie informacji dotyczącej stosowanego formatu danych, przyporządkowania pól w ramce do konkretnych czujników itp. W tej sytuacji konieczne jest wykorzystanie metod, które pozwolą na określenie typowej treści w procesie automatycznego uczenia bez potrzeby korzystania z informacji podanych przez projektanta.

Metodą wybraną przez autora jest analiza n-gramów. Takie podejście było proponowane jako metoda wykrywania rozprzestrzeniania się robaków w sieci Internet [11]. N-gram jest definiowany jako n-elementowa sekwencja elementów (w tym przypadku bajtów) zawarta w przesyłanych danych. Udział poszczególnych n-gramów w całości komunikacji jest miarą charakterystyczną dla typowego ruchu, w stosunku do której można odnieść ruch obserwowany aktualnie. Ponieważ liczba K koniecznych do określenia n-gramów gwałtownie rośnie wraz ze wzrostem n, $K=256^n$, w praktyce często przyjmuje się n=1, co sprowadza

podejście do określenia względnej częstości występowania wszystkich możliwych wartości pojedynczego bajtu; takie też rozwiązanie wykorzystał autor.

Ponownie w fazie uczenia określany jest typowy rozkład n-gramów, który następnie jest wykorzystywany do tworzenia sieci bayesowskiej.

4.4. Łączenie źródeł danych – sieć bayesowska

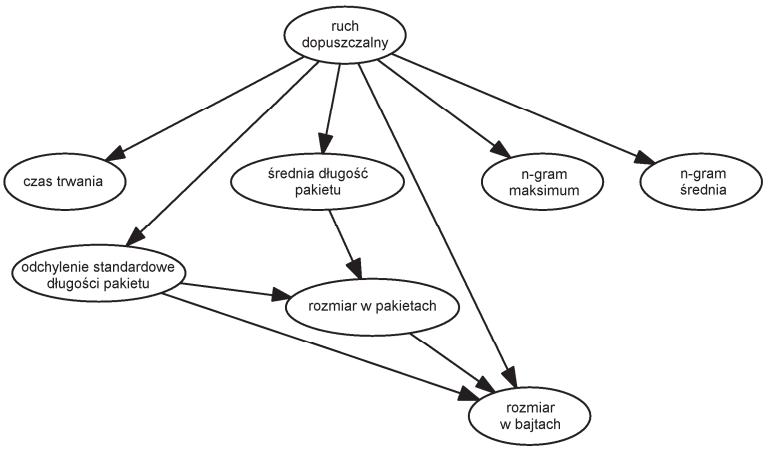
Po przeprowadzeniu analiz opisanych w poprzednich dwu podpunktach system dysponuje dla każdej transmisji zbiorem wyników mogących świadczyć o anomalii. Konieczne jest połączenie tych przesłanek, aby uzyskać pojedynczą wartość, która może być łatwo zinterpretowana. W tym celu wykorzystywana jest sieć bayesowska. Proces konstruowania sieci przebiega następująco:

- w pierwszym etapie, na podstawie obserwowanego ruchu sieciowego, określane są przedziały wartości opisanych w podpunkcie 4.1 oraz uśrednione rozkłady 1-gramów – te ostatnie osobno dla ramek o różnych długościach,
- w drugim etapie określany jest rozrzut wartości n-gramów dla normalnego ruchu: celem jest określenie, jak bardzo w normalnym ruchu n-gramy mogą różnić się od wartości średniej,
- w trzecim etapie zebrane dane są poddawane procesowi dyskretyzacji i stanowią podstawę do uczenia struktury i parametrów sieci.

W powyższym opisie zwraca uwagę oparcie całego procesu uczenia wyłącznie na danych reprezentujących normalny ruch. W ogólności należy bowiem przyjąć, że dane reprezentujące atak (bądź innego rodzaju anomalię) nie będą dostępne. Tym samym jest to problem określany jako *one-class classification*. Wykorzystane algorytmy uczenia nie są dostosowane do przeprowadzania uczenia w przypadku dostępności danych należących tylko do jednej klasy. Aby umożliwić uczenie, dane są uzupełniane przez próbki wygenerowane przez losowy wybór wartości. Stanowią one „szum”, w stosunku do którego algorytmy uczące mogą określić kombinacje wartości charakterystyczne dla ruchu normalnego.

Tak skonstruowana sieć może być następnie wykorzystana do określenia prawdopodobieństwa przynależności obserwowanego ruchu do klasy ruchu dopuszczalnego. Próg detekcji można wyznaczyć na podstawie wartości prawdopodobieństw wyliczonych dla ruchu normalnego.

Przykładowa struktura wygenerowanej sieci jest przedstawiona na rys. 1.



Rys. 1. Przykładowa struktura sieci bayesowskiej
Fig. 1. Sample structure of the Bayesian network
Źródło: Tylman W.: Threat Detection Systems Using Bayesian Networks, Politechnika Łódzka, 2013.

W ramach testów zaprezentowanego podejścia generowano wymiany danych między dwoma urządzeniami; wymiany charakteryzowały się zmiennym stopniem zróżnicowania formy i treści. System był w stanie wychwycić wymiany odbiegające od typowych, przy czym próg czułości był uzależniony od stopnia zróżnicowania występującego podczas fazy uczenia – takie zachowanie jest zachowaniem pożądanym.

4.5. Modelowanie obiektu sterowanego – sieć MEBN

Jako uzupełnienie rozwiązania opisanego w podpunktach 4.1-4.4 autor zaproponował utworzenie probabilistycznego modelu obiektu sterowanego, na podstawie którego można wykryć zachowania obiektu odbiegające od normy. Jako formalizm opisu autor wybrał sieci MEBN. Ponieważ obecnie sieci te nie mają wsparcia dla uczenia z danych, opis musi być utworzony przez eksperta.

Jako przykładowy obiekt sterowany wybrano prosty piec, wyposażony w kilka czujników i elementów wykonawczych, przedstawionych w tabeli 1.

Tabela 1
Czujniki i elementy wykonawcze przykładowego obiektu sterowanego

Nazwa	Opis
T1-T3	czujniki temperatury w trzech miejscach pieca
R	czujnik obrotów silnika przesuwu produktów do wypalenia
C	czujnik zawartości CO_2
A	sterowanie silnikiem nadmuchu powietrza do komory spalania
F	sterowanie silnikiem podawania paliwa do komory spalania
E	sterowanie silnikiem przesuwu produktów
M	sterowanie silnikiem podawania substratów
I	sterowanie urządzeniem zapłonowym

W tworzeniu modelu wykorzystano zależności pomiędzy stanami czujników i elementów wykonawczych wynikające z przyjętego algorytmu sterowania, a także zależności wynikające z fizycznych ograniczeń zachodzących procesów (np. związki między temperaturami mierzonymi przez poszczególne czujniki). Zależności te pozwoliły na utworzenie opisu składającego się z 10 elementów MFrag (elementów składowych opisu MEBN). W stosunku do klasycznych sieci bayesowskich opis MEBN pozwolił na wygodne zakodowanie zależności logicznych oraz na utworzenie jednego opisu działającego z dowolna liczbą czujników temperatury. Należy jednak zauważyć, że utworzony opis okazał się złożony, co przy braku wsparcia dla uczenia maszynowego przełożyło się na czasochłonność jego konstruowania. Również czas obliczeń w programie UnBBayes był bardzo długi (maksymalnie do 100 minut), co uniemożliwia wykorzystanie podejścia w praktyce. Ta ostatnia wada wydaje się jednak związana z działaniem programu UnBBayes, a nie ze specyfiką sieci MEBN.

5. Podsumowanie

Autor zaprojektował i wykonał system detekcji anomalii oparty na sieciach bayesowskich i wykorzystujący oprogramowanie dotychczas stosowane w sieci Internet. Wstępne wyniki wskazują na poprawność przyjętych założeń. Klasyczne sieci bayesowskie sprawdziły się jako metoda łączenia wyników poszczególnych analiz. Jednocześnie zastosowanie sieci MEBN w obecnym stanie rozwoju oprogramowania do ich symulacji wydaje się nie mieć uzasadnienia ze względu na złożony proces tworzenia opisu i nieakceptowalny czas obliczeń.

Prace przedstawione w tym artykule były finansowane przez Narodowe Centrum Nauki w drodze grantu badawczego (umowa nr 4769/B/T02/2011/40). Prezentowany system wykorzystuje bibliotekę SMILE, a rys. 1 utworzono za pomocą interfejsu graficznego GeNIe – oba te programy zostały opracowane przez Decision Systems Laboratory, University of Pittsburgh i są dostępne pod adresem http://genie.sis.pitt.edu.

Bibliografia

1. Betta G., Pietrosanto A.: Instrument fault detection and isolation: state of the art and new research trends. „IEEE Trans. on Instrumentation and Measurement”, Vol. 49, No. 1, 2000, p. 100–107.

2. Bigham J., Gamez D., Lu N.: Safeguarding SCADA systems with anomaly detection. Proceedings of the Second International Workshop on Mathematical Methods, Models and Architectures for Computer Network Security MMMACNS 2003, Lecture Notes in Computer Science. Springer Verlag, 2003, p. 171–182.
3. Cheung S., Dutertre B., Fong M., Lindqvist U., Skinner K., Valdes A.: Using model-based intrusion detection for SCADA networks. Proceedings of the SCADA Security Scientific Symposium, 2007, p. 1–12.
4. Holbert K.E., Mishra A., Mili L.: Intrusion detection through SCADA systems using fuzzy logic-based state estimation methods. „Int. J. of Critical Infrastructures", No. 3(1/2), 2007, p. 58–87.
5. Laskey K.B.: MEBN: A language for first-order Bayesian knowledge bases. „Artificial Intelligence", No. 172, 2008, p. 140–178.
6. Linda O., Vollmer T., Manic M.: Neural network based intrusion detection system for critical infrastructures. Proceedings of the International Joint Conference on Neural Networks, 2009, p. 1827–1834.
7. Pearl J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann Publishers, San Mateo 1988.
8. Sommestad T., Ekstedt M., Johnson P.: Cyber security risks assessment with Bayesian defense graphs and architectural models. Proceedings of the Hawaii International Conference on System Sciences, 2009, p. 1–10.
9. Valdes A., Cheung S.: Communication pattern anomaly detection in process control systems. Proceedings of the IEEE Conference on Technologies for Homeland Security. IEEE, 2009, p. 22–29.
10. Valdes A., Skinner K.: Adaptive, model-based monitoring for cyber attack detection. Recent Advances in Intrusion Detection, Lecture Notes in Computer Science. Springer, Berlin 2000, p. 80–92.
11. Wang K., Stolfo S.J.: Anomalous payload-based network intrusion detection. Recent Advances in Intrusion Detection, Lecture Notes in Computer Science. Springer, Berlin 2004, p. 203–222.
12. Snort project, http://www.snort.org. Dostęp: czerwiec 2014.
13. UnBBayes project, http://unbbayes.sourceforge.net. Dostęp: czerwiec 2014.
14. Viking project, http://www.vikingproject.eu. Dostęp: czerwiec 2014.

Abstract

The paper presents a concept of a highly automated solution allowing detection, in industrial network traffic, of situations differing from the normal state (anomalies). The solution is based on modelling the allowable communication patterns, allowable communication content and behaviour of the controlled objects. It uses classical Bayesian networks to fuse partial results coming from different sources and Multi-Entity Bayesian Networks (MEBN) to model the controlled objects. The work also illustrates the possibility of using existing software (taking Snort system as an example) and the required modifications related to the support for non-IP networks (taking Modbus RTU protocol as an example).