Rafał KOZIK, Michał CHORAŚ
ITTI Sp. z o.o., Rubież 46, Poznań
INSTYTUT TELEKOMUNIKACJI, UNIWERSYTET TECHNOLOGICZNO PRZYRODNICZY, Kaliskiego 7, Bydgoszcz

Koncepcja wymiany informacji w systemie ochrony sieci federacyjnych

Mgr inż. Rafał KOZIK

Pracownik naukowo-dydaktyczny zatrudniony na stanowisku asystenta w Zakładzie Komunikacji Komputerowej na Uniwersytecie Technologiczno Przyrodniczym w Bydgoszczy. Jest autorem i współautorem ponad 40 publikacji naukowych z dziedziny bezpieczeństwa sieci teleinformatycznych, biometrycznej identyfikacji osób oraz komputerowej wizji.

e-mail: rafal.kozik@utp.edu.pl

Dr inż. Michał CHORAŚ

Adiunkt w Instytucie Telekomunikacji UTP Bydgoszcz oraz starszy konsultant i kierownik projektów w ITTI Sp. z o. o. Jest autorem ponad 100 publikacji naukowych z zakresu przetwarzania obrazów, biometrycznego rozpoznawania osób, ochrony krytycznych infrastruktur oraz bezpieczeństwa sieci teleinformatycznych.

e-mail: chorasm@utp.edu.pl

Streszczenie

W niniejszym artykule przedstawiono koncepcję wymiany informacji pomiędzy domenami w systemie ochrony sieci federacyjnych. Zaproponowano architekturę systemu ochrony sieci federacyjnych, a w szczególności omówiono zasadę działania Modułów Decyzyjnych (MD). Zaprezentowany sposób komunikacji między domenami wykorzystuje technologię P2P (Peer to Peer). Moduły Decyzyjne w poszczególnych domenach federacji mogą wymieniać informacje o stanie sieci, wykrytych działaniach nieuprawnionych oraz Ogólne Reguły Decyzyjne (ORD) będące wypracowanymi poleceniami reakcji. Współpraca domen w federacji pozwala na osiągnięcie efektu synergii i zwiększenie bezpieczeństwa sieci (m.in. sieci wykorzystywanych w administracji publicznej lub sieci militarnych). Opisano kwestie związane z bezpieczeństwem technologii P2P oraz przedstawiono scenariusz ukazujący korzyści płynące z zaproponowanego rozwiązania. Przedstawiona koncepcja jest rezultatem prac w projekcie rozwojowym SOPAS finansowanym przez MNiSW w zakresie bezpieczeństwa państwa.

Słowa kluczowe: bezpieczeństwo sieci komputerowych, sieci federacyjne, komunikacja P2P.

Information exchange between domains in the Federated Networks Protection System

Abstract

In this paper a concept and architecture of the Federated Networks Protection System (FNPS) are presented. The system components and, particularly, the Decision Module are described. The major contribution of the paper is the concept of P2P (Peer to Peer) based information exchange between federated networks. Communication between Decision Modules (DM) in each of the federated domain is based on P2P in order to inform about network status, detected attacks or anomalies and distribute General Decision Rules (GDR) describing specific reactions. The presented system is dedicated for federated networks and systems used by the public administration and military sector. Such systems can increase their overall security and resiliency by sharing and exchanging security related information and general reaction rules. There is also presented a sample scenario (SQLIA – SQL injection attack detection) to show how the proposed system can detect complex attacks and benefit from information sharing between federated domains.

Keywords: network security, federated networks, peer-to-peer communication.

1. Wprowadzenie

Obecnie, po przeprowadzeniu skutecznych ataków sieciowych (tzw. cyber ataków) na państwa takie jak Estonia, Gruzja, Iran oraz na duże korporacje, w tym Google oraz Sony, cyber ataki są uznawane za jedno z większych zagrożeń dla krytycznych infrastruktur (np. gridów energetycznych) oraz bezpieczeństwa państwa (np. system finansowy) [1]. Na przykład w 2008 roku przeprowadzono skuteczny atak DDoS (*Distributed Denial of Service*) na strony gruzińskiego rządu, prezydenta Gruzji oraz Narodowy Bank Gruzji [1]. Ataki sieciowe są poważnym zagrożeniem dla sieci administracji publicznej oraz sieci wojskowych.

W niniejszym artykule przedstawiono wstępne wyniki projektu rozwojowego „System ochrony sieci teleinformatycznych przed działaniami nieuprawnionymi (SOPAS)", który jest finansowany przez MNiSW w ramach tematu bezpieczeństwo państwa. Celem systemu ochrony sieci federacyjnych SOPAS jest ochrona sieci administracji publicznej oraz sieci wojskowych, które często połączone są w tzw. federację (*Federations of Systems* (FoS)). Dzięki zastosowaniu podejścia zgodnego z federacją systemów (sieci), można osiągnąć efekt synergii oraz zwiększyć bezpieczeństwo federacji.

W niniejszym artykule, pokazano możliwości sieci federacyjnych do współdzielenia i wymiany informacji dotyczących bezpieczeństwa, takich jak zdarzenia w sieci, wykryte działania nieuprawnione oraz proponowane środki zaradcze. Podejście polegające na współdzieleniu informacji dot. bezpieczeństwa pomiędzy zaufanymi domenami jest zgodne z ideą FoS. W proponowanym podejściu, domeny wchodzące w skład federacji nie są centralnie zarządzane, ale współpracują ze sobą. Takie rozwiązania mogą wkrótce zastąpić nieefektywne podejście tzw. "zamkniętego bezpieczeństwa" [2].

Niniejszy artykuł ma następujący układ: w sekcji 2 przedstawiono ogólną architekturę systemu ochrony sieci federacyjnych zaproponowaną w projekcie SOPAS. W sekcji 3 przedstawiono propozycję wymiany informacji między domenami opartej o P2P. W sekcji 4 przedstawiono analizę rozwiązania opartego o P2P pod kątem bezpieczeństwa. Natomiast w sekcji 5 opisano scenariusz obrazujący korzyści z wymiany informacji między domenami na przykładzie ochrony przed atakami typu SQLIA.

2. System ochrony sieci federacyjnych

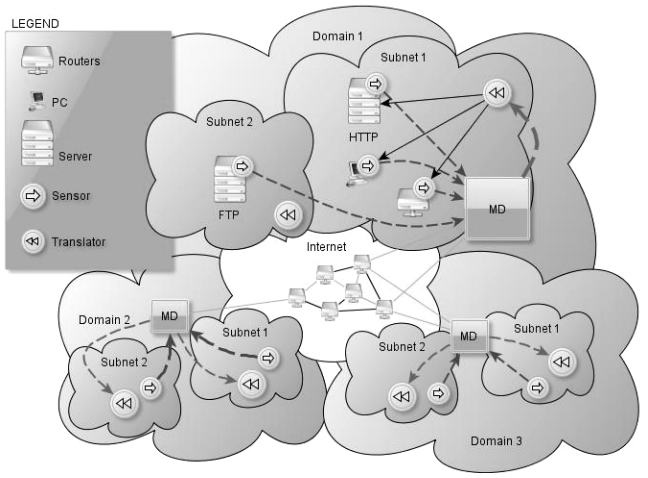
System ochrony sieci federacyjnych składa się z połączonych niezależnych domen. W każdej domenie znajdują się sensory, Moduł Decyzyjny (MD) oraz elementy reakcji (rys. 1) [3].

W systemie ochrony sieci federacyjnych wykorzystywane są typowe sensory już wcześniej zainstalowane w domenach, a także, o ile jest to możliwe dodatkowe sensory, takie jak system ARAKIS [4], system HSN [5], SNORT z dodatkowymi pre-procesorami SOPAS (np. system ADS) itp. Podobnie, wykorzystywane są dostępne elementy reakcji, w tym firewalle, pułapki itp.

Moduł Decyzyjny powinien być umieszczony (logicznie) w każdej domenie federacji sieci chronionych przez system SOPAS. Na rys. 1 pokazano umiejscowienie Modułów Decyzyjnych w domenach federacji.

Kluczowym zadaniem Modułu Decyzyjnego jest analiza danych otrzymanych z elementów sensorycznych. Informacje te wraz z wiedzą na temat bezpieczeństwa sieciowego (opisanej w postaci ontologii) wykorzystane są w procesie korelacji w celu podjęcia decyzji o wykryciu wystąpienia (bądź nie) działań nieuprawnio-

nych [3]. MD tworzy i udostępnia elementom reakcji ogólną regułę decyzyjną (ORD).



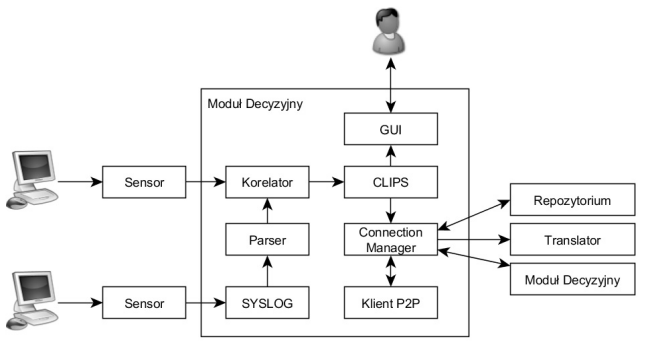
Rys. 1. Ogólna architektura federacji systemów. Na Rysunku Moduł Decyzyjny oznaczono jako MD
Fig. 1. General architecture of the Federated Networks Protection System (FNPS)

Ogólna architektura Modułu Decyzyjnego jest przedstawiona na rys. 2. MD składa się z interfejsów wej/wyj oraz następujących komponentów:
- Korelator (oparty o system Borealis) [6],
- Silnik reguł CLIPS,
- Graficzny interfejs użytkownika,
- SYSLOG – systemowy daemon zarządzający logami,
- Connection Manager.

MD posiada interfejs wejściowy odpowiedzialny za otrzymywanie strumieni danych z sensorów oraz interfejs wyjściowy odpowiedzialny za przesłanie Ogólnych Reguł Decyzyjnych do elementów reakcji.

MD komunikuje się z innymi MD umieszczonymi w innych domenach wykorzystując technologię P2P (klient P2P na rys. 2).

MD komunikuje się również z Repozytorium poprzez protokół SOAP.



Rys. 2. Architektura Modułu Decyzyjnego
Fig. 2. Decision Module architecture and components

Strumienie zdarzeń mogą być wysyłane do Korelatora w MD bezpośrednio, albo poprzez proces SYSLOG.

MD posiada również GUI, które pozwala operatorowi na obserwowanie aktualnie wypracowywanych reguł decyzyjnych, wykrytych działań nieuprawnionych oraz odebranych zdarzeń pośrednich z Korelatora.

Przedstawiony na rys. 2 element o nazwie Connection Manager odpowiedzialny jest za nadzorowanie komunikacji z innymi MD (poprzez klienta P2P), elementami reakcji oraz repozytorium (przechowującym wypracowane ORD).

Poszczególne elementy architektury MD oraz mechanizmy korelacji informacji w celu podejmowania decyzji omówione zostały w [3].

3. Wymiana informacji między domenami z wykorzystaniem P2P

W systemie SOPAS, Moduły Decyzyjne przesyłają sobie informacje o zdarzeniach w sieci (tzw. „fakty") oraz (gdy wykryto atak) wypracowane Ogólne Reguły Decyzyjne (ORD) zawierające symptom oraz regułę (środek zaradczy).

W niniejszej sekcji przedstawiono propozycję oraz realizację wymiany informacji między domenami w oparciu o technologię P2P. Natomiast sposób korelacji informacji z sensorów, proces decyzyjny oraz sposób wypracowywania ORD nie są omówione w niniejszym artykule.

Peer-to-peer (P2P) to technologia, która pozwala rozproszyć zasoby (dane, moc obliczeniową, miejsce na dysku) pomiędzy jednakowo uprzywilejowane maszyny (aplikacje).

W sieci P2P pojedynczy element (*peer*) udostępnia część swoich zasobów innym uczestnikom biorącym udział w komunikacji. Każdy peer w trakcie transmisji może jednocześnie pełnić rolę dostawcy lub konsumenta zasobów.

Systemy wykorzystujące P2P tworzą abstrakcyjną sieć powiązań (*overlay network*) pomiędzy poszczególnymi elementami biorącymi udział w transmisji. Takie podejście pozwala na uniezależnienie się od faktycznego fizycznego połączenia pomiędzy elementami.

W sieci P2P każdy klient (*peer*) dostarcza zasoby sieciowe (pasmo, miejsce na dysku, moc obliczeniową). W momencie przyłączania się do sieci nowych klientów, całkowita pojemność systemu rośnie. Jest to zatem rozwiązanie całkowicie inne niż klasyczne podejście klient-serwer, w którym wraz z przybywaniem nowych klientów zasoby serwerowe maleją.

Rozproszona architektura systemów P2P pozwala na osiągnięcie dużej niezawodności w dostępie do zasobów. Wynika to z możliwości zmiany ścieżki pomiędzy dwoma elementami w sposób dynamiczny.

W przypadku uszkodzenia jednego lub wielu elementów w sieci system może ciągle pełnić swoje funkcje, gdyż każdy peer biorący udział w transmisji jest jednakowo ważny. W klasycznym rozwiązaniu klient-serwer, uszkodzenie serwera powoduje zatrzymanie usług świadczonych przez system.

Technologie P2P, w zależności od sposobu tworzenia topologii sieci, dzielone są na trzy grupy rozwiązań:
1. Structured P2P,
2. Unstructured P2P,
3. Hybrid P2P.

Ze względu na specyfikę architektury systemu ochrony sieci federacyjnych SOPAS, gdzie istnieje wiele równorzędnych domen, proponowanym rozwiązaniem jest technologia P2P typu unstructured. Implementacja mechanizmu P2P w projekcie SOPAS przypomina system Gnutella v0.4 (rozwiązanie typowo unstructured). Rozwiązanie takie pozwala na łatwe i dynamiczne rozbudowanie systemu wzajemnie połączonych domen. Przy niewielkiej ilości danych jakie są przesyłane pomiędzy MD (ogólne reguły decyzyjne), takie rozwiązanie nie generuje dużego ruchu związanego z sygnalizacją.

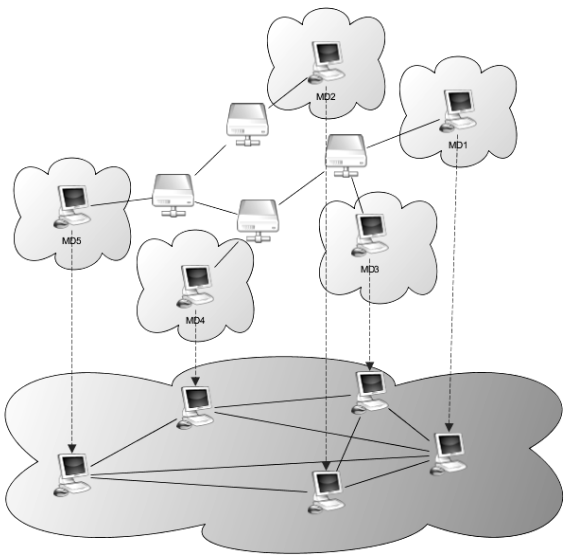
Dodatkowo, ze względu na to, iż system ochrony sieci federacyjnych SOPAS wykorzystuje dwukierunkowy sposób transmisji (nie tylko pobieranie danych z sieci, ale także ich wysyłanie), rozwiązanie oparte o sieci typu unstructured jest uzasadnione.

Ogólna wizja wykorzystania sieci P2P w architekturze systemu ochrony sieci federacyjnych SOPAS pokazana została na rys. 3.

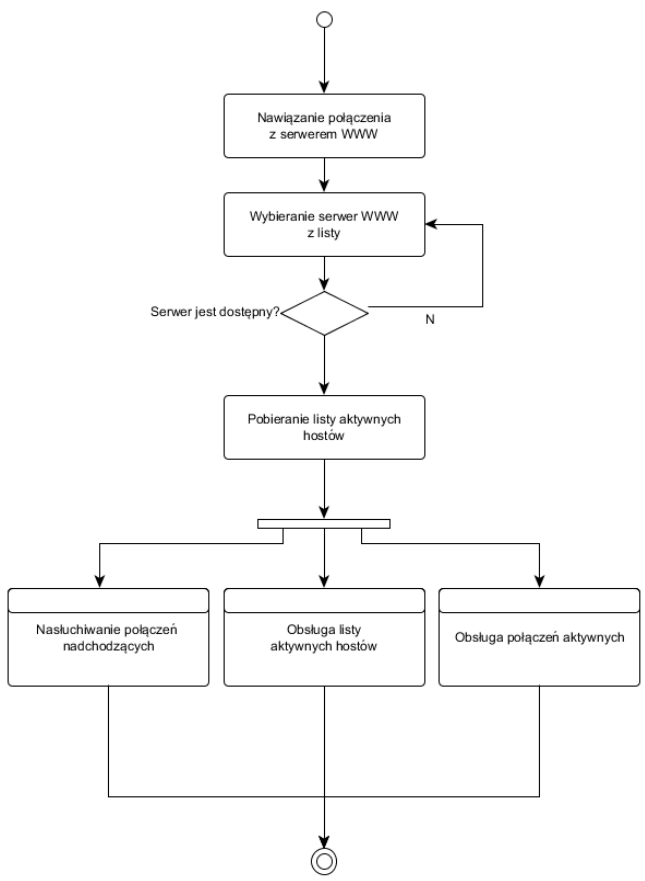
Moduły Decyzyjne każdej domeny w federacji sieci w systemie SOPAS są połączone *P2P Overlay Network* (Gnutella v 0.4).

Moduły Decyzyjne wysyłają oraz pobierają wypracowane w procesie decyzyjnym ogólne reguły decyzyjne (ORD) do/z innych Modułów Decyzyjnych w federacji [3]. Rozwiązanie takie pozwala na komunikację między Modułami Decyzyjnymi, a także na replikację danych oraz replikację ścieżek. Diagram aktywności klienta P2P przedstawiający proces inicjalizacji i pracy pokazany został na rys. 4. Proces inicjalizacji obywa się w wątku głównym programu. Po zaczytaniu listy aktywnych hostów z serwisu

WWW (inicjalizacja przez serwer WWW) uruchomione zostają trzy równoległe wątki (niebieskie bloki na rys. 4).



Rys. 3. Propozycja wykorzystania *P2P overlay network* w projekcie SOPAS
Fig. 3. The concept of using P2P in the Federated Networks Protection System



Rys. 4. Diagram aktywności klienta P2P
Fig. 4. P2P client activity diagram

Aby możliwa była obsługa wielu żądań równolegle, każde nowopowstałe połączenie jest obsługiwane w osobnym wątku.

Aktualna implementacja komunikacji P2P specyfikuje następujące rodzaje pakietów sygnalizacyjnych:

- Pakiet ping-pong, służący do walidacji czy zestawiono połączenie z właściwym klientem P2P.
- Pakiet zawierający listę aktywnych hostów od innego klienta P2P.
- Pakiet żądania przesłania listy aktywnych hostów, otrzymany od innego klienta P2P.
- Pakiet zawierający ORD.

Aktualna implementacja komunikacji P2P specyfikuje dwa rodzaje list hostów:

- Lista aktywnych hostów, zawierająca aktywne połączenia.
- Lista znanych hostów, zawierająca adresy IP, które potencjalnie mogą być aktywne.

Lista znanych hostów jest gromadzona automatycznie przez klienta P2P. W trakcie pierwszego uruchomienia lista ta zawiera wyłącznie adresy pozyskane z serwera WWW. W trakcie dalszej pracy lista ta rozrasta się o nowe adresy i jest zapisywana przez program w pamięci lokalnej. Każdy pakiet odbierany poprzez aktywne połączenia jest walidowany pod kątem poprawności. Walidacja obejmuje:

- Sprawdzenie czasu życia pakietu (pakiet TTL).
- Detekcję duplikatów (pakiet timestamp) w oknie czasowym.

4. Analiza bezpieczeństwa komunikacji z wykorzystaniem P2P

Komunikacja zakłada istnienie wielu równorzędnych klientów P2P, który w każdej chwili może pełnić role klienta jak i serwera pośredniczącego w komunikacji pomiędzy innymi klientami. W takiej sytuacji stosunkowo łatwo jest atakującemu przeprowadzić szereg działań, które mogą zaburzyć poprawne działanie komunikacji pomiędzy modułami decyzyjnymi. Takimi zagrożeniami są:

- Możliwość podsłuchu.
- Możliwość modyfikacji zawartości pakietów przesyłanych pomiędzy klientem A oraz B (atak *man-in-the-middle*).
- Możliwość podszycia się pod MD i generowania fałszywych ORD do innych MD.
- Możliwość przechwytywania pakietów i ich niszczenia.

Dlatego też, by zabezpieczyć się przed problemami podsłuchu zawartości pakietu jaki i jego modyfikacji zaproponowano wykorzystanie komunikacji z wykorzystaniem SSL. Domeny, między którymi przebiegać będzie komunikacja, muszą sobie ufać, a co za tym idzie muszą potwierdzić swoją tożsamość. W tym celu zaproponowano użycie certyfikatów. Wymaga to utrzymywania CA (*Certificate Authority*) wewnątrz systemu SOPAS.

Od nowo przyłączonego Modułu Decyzyjnego do systemu SOPAS wymagać się będzie wygenerowania klucza publicznego oraz podpisania go przez CA. Pozwoli to na uniknięcie, w dużym stopniu, problemów powiązanych z podszywaniem się atakującego pod MD oraz przechwytywania przez niego pakietów.

5. Przykładowy scenariusz wymiany informacji (atak typu SQLIA)

Celem zaprezentowanego scenariusza jest wykrycie ataku typu SQLIA i pokazanie, że korelacja wielu strumieni zdarzeń, ich analiza w Module Decyzyjnym oraz wymiana informacji w federacji sieci działa skuteczniej niż analiza informacji z pojedynczych sensorów tylko w pojedynczych domenach.

Scenariusz ataku SQLIA pokazuje jak słabo chronione domeny mogą czerpać korzyść z współdzielenia informacji w federacji.

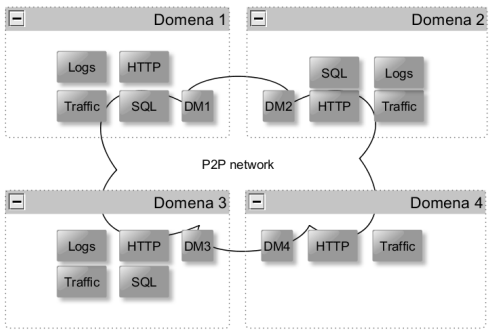
Ataki typu SQLIA (*SQL Injection Attack*) jest sklasyfikowany na pierwszym miejscu listy zagrożeń według Open Web Application Security Project (OWASP) (lista *The Ten Most Critical Web Application Security Risks*) [10].

Zakładamy istnienie 4 domen i 4 MD (Rys. 5). Domeny są w swojej grupie domenowej („ufają sobie"). Mają różne polityki, więc mogą mieć zdefiniowane różne reakcje na te same ataki. Komunikacja między MD odbywa się w oparciu o P2P.

Informacje, na temat „co" może być wysłane i „do którego MD" pobierane są z Repozytorium każdego MD (ten krok pominięto w poniższym scenariuszu).

W zaproponowanym scenariuszu wykorzystywane są następujące sensory:

a) SNORT (analizuje ruch w domenie – warstwa transportowa + warstwa aplikacji).
b) Analizator logów warstwy aplikacji (np. skrypty shellowe).
c) Analizator SEC (Simple Event Correlator).
d) Moduły Decyzyjne z innych domen mogą być także potraktowane jako sensory (przekazują informacje o zdarzeniach tzw. "fakty" oraz wypracowane ORD).



Rys. 5. Topologia sieci wykorzystywanej w scenariuszu ochrony przed atakami typu SQLIA
Fig. 5. Simplified topology diagram of the federation (orange box - decision module, green box - hosted services, blue box - installed sensors)

Poszczególne kroki scenariusza są następujące:

1. Atakujący skanuje adresy w poszukiwaniu serwisów WWW wewnątrz domen.
2. Sensory (np. SNORT) wychwytują próbę skanowania i wysyłają informację do swoich MD (załóżmy że SNORT jest w domenie 2 i 3, więc tą informację mają MD2 i MD3).
3. MD2 i MD3 zapamiętują adres sieciowy atakującego.
4. Atakujący wykonuje równolegle testy penetracyjne wielu serwisów w poszukiwaniu luk bezpieczeństwa.
5. Sensory wykrywają w logach http wzmożony ruch pochodzący z jednego adresu IP. Sensory przesyłają zdarzenie do MD w swojej domenie (załóżmy że tylko w domenie 1 więc tą informację ma MD1 i rozsyła tą informację („fakt") do innych MD).
6. Sensory wykrywają w logach SQL szereg nieudanych zapytań SQL. Sensory przesyłają zdarzenie do MD w swojej domenie (załóżmy że tylko w domenie 1 więc tą informację ma MD1 i rozsyła tą informację („fakt") do innych MD).
7. Sensory warstwy transportowej w jednej z domen (mało uczęszczanej przez użytkowników) alarmują o wzmożonym ruchu generowanym przez atakującego. Informacja zostaje przesłana do innych MD. Zakładamy, że w tej domenie atakowany serwis nie ma zainstalowanych sensorów http i SQL (np. jest to serwis mniej krytyczny) (niech będzie to domena 4 więc tą informację ma MD4 i rozsyła tą informację „fakt" do innych MD).
8. W domenach, gdzie atak się powiódł sensory w warstwie transportowej wychwytują duże ilości danych pobieranych z domeny (załóżmy że tylko w domenie 3 więc tą informację ma MD3 i rozsyła tą informację „fakt" do innych MD).
9. Domeny po wymianie informacji posiadają dane o zdarzeniach
   a. Skanowania
   b. Wzmożonym ruchu w warstwie transportowej
   c. Wzmożonym ruchu w warstwie aplikacji (http, SQL)
10. MD1 na podstawie ontologii wypracowuje ORD (reakcją jest „powiadom administratora").
11. MD2 na podstawie ontologii wypracowuje ORD . Dla MD2, ORD nakazuje blokowanie ruchu z danego IP na 15min.
12. MD2 wysyła ORD do odpowiednich i dostępnych Elementów Reakcji (ER). Ruch z danego adresu IP jest blokowany.
13. MD1 i MD2 przesyłają „fakt" o ataku do MD3 i MD4.
14. MD3 i MD4 na podstawie „faktu" z MD1 i MD2 wypracowują ORD dla ER. Ruch z danego adresu IP jest blokowany.

6. Podsumowanie

W niniejszym artykule przedstawiono ogólną koncepcję oraz architekturę systemu ochrony sieci federacyjnych. Następnie zaprezentowano sposób wymiany informacji między domenami wykorzystujący technologię P2P.

Opisano także kwestie związane z bezpieczeństwem P2P oraz przedstawiono scenariusz ukazujący korzyści zaproponowanego rozwiązania.

Niniejszy artykuł częściowo sfinansowano ze środków projektu rozwojowego „System ochrony sieci teleinformatycznych przed działaniami nieuprawnionymi" (0 R00 0125 11).

7. Literatura

[1] Enabling and managing end-to-end resilience, ENISA (European Network and Information Security Agency) Report, January 2011.
[2] Choraś M., D'Antonio S., Kozik R., Hołubowicz W.: INTERSECTION Approach to Vulnerability Handling , In Proc. of WEBIST 2010, vol. 1, 171-174, INSTICC Press, Valencia, Spain, April 2010.
[3] Choraś M, Kozik R., Piotrowski R., Brzostek J., Hołubowicz W.: Network Events Correlation for Federated Networks Protection System, In Abramowicz W et al. (Eds).: Towards a Service Based Internet, LNCS, Springer-Verlag, 2011.
[4] Strona ARAKIS: http://www.arakis.pl.
[5] Strona HSN: http://www.honeyspider.net/.
[6] Strona Borealis: http://www.cs.brown.edu/research/borealis/public.
[7] NATO Network Enabled Feasibility Study Volume II: Detailed Report Covering a Strategy and Roadmap for Realizing an NNEC Networking and Information Infrastructure (NII), version 2.0.
[8] El-Damhougy, Yousefizadeh H., Lofquist H., Sackman D., Crowley R.: Hierarchical and federated network management for tactical environments, In Proc. of IEEE Military Communications Conference MILCOM, vol. 4, 2062 – 2067, 2005.
[9] Calo S., Wood D., Zerfos P., Vyvyan D., Dantressangle P., Bent G.: Technologies for Federation and Interoperation of Coalition Networks, In Proc. of 12th International Conference on Information Fusion, Seattle, 2009.
[10] OWASP Top Ten - 2010. The Ten Most Critical Web Application Security Risks. Published by Open Web Application Security Project (OWASP).
[11] Pallickara Shrideep et al.: A Framework for Secure End-to-End Delivery of Messages in Publish/Subscribe Systems. IEEE Computer Society, 215-222, 2006.
[12] Beitollahi H. and Deconinck G.: Analyzing the Chord Peer-to-Peer Network for Power Grid Applications. 2008.
[13] Stoica Ion et al., Chord: A scalable peer-to-peer lookup service for internet applications. SIGCOMM '01: Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications, ACM, 149-160, 2001.
[14] Deconinck G. et al.: A Robust Semantic Overlay Network for Microgrid Control Applications. Chapter in R. De Lemos, F. Di Giandomenico, C. Gacek, H. Muccini, M. Vieira (Eds.), Architecting Dependable Systems V, 101-123, 2008.
[15] Maymounkov P., Mazieres D., Kademlia: A Peer-to-Peer Information System Based on the XOR Metric., Springer-Verlag, 53-65, 2002.
[16] Cohen E. and Shenker S.: Replication Strategies in Unstructured Peer-to-Peer Networks. ACM, 2002. SIGCOMM '02: Proceedings of the 2002 Conference on Applications, Technologies, Architectures, and Protocols for computer Communications. Vol. 32, pp. 177-190, 2002.
[17] Antony I. and Druschel P., Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems, Springer-Verlag, 329-350, 2001.

otrzymano / received: 03.11.2011
przyjęto do druku / accepted: 03.01.2012 *artykuł recenzowany / revised paper*