© Tereykovsky I., Korchenko A., Parashchuk T., Pedchenko Y., Open intrusion detection systems analysis // Ukrainian Scientific Journal of Information Security, 2018, vol. 24, issue 3, pp. 201-216.

УПРАВЛІННЯ ІНФОРМАЦІЙНОЮ БЕЗПЕКОЮ / INFORMATION SECURITY MANAGEMENT

DOI: 10.18372/2225-5036.24.13431

АНАЛІЗ ВІДКРИТИХ СИСТЕМ ВИЯВЛЕННЯ ВТОРГНЕНЬ

**Ігор Терейковський, Анна Корченко,
Тарас Паращук, Євгеній Педченко**

Національний авіаційний університет, Україна

ТЕРЕЙКОВСЬКИЙ Ігор Анатолійович, д.т.н., професор

Рік і місце народження: 1967 рік, м Тернопіль, Україна.
Освіта: Київський інститут інженерів цивільної авіації, 1992 рік.
Посада: професор кафедри системного програмування і спеціалізованих комп'ютерних систем НТУУ «КПІ ім. Ігоря Сікорського» з 2015 року.
Наукові інтереси: інформаційна безпека.
Публікації: більше 100 наукових праць, серед яких монографії, навчальні посібники, навчально-методичні комплекси дисциплін, наукові статті.
E-mail: terejkowski@ukr.net

КОРЧЕНКО Анна Олександрівна, к.т.н., доцент

Рік і місце народження: 1985 рік, м. Київ, Україна.
Освіта: Національний авіаційний університет, 2007 рік.
Посада: доцент кафедри безпеки інформаційних технологій.
Наукові інтереси: інформаційна безпека, системи виявлення вторгнень, експертне оцінювання в сфері захисту інформації.
Публікації: більше 90 наукових публікацій, серед яких наукові статті, підручники та навчально-методичні посібники.
E-mail: annakor@ukr.net

ПАРАЩУК Тарас Іванович

Рік і місце народження: 1996 рік, м. Вінниця, Україна.
Освіта: Національний авіаційний університет, 2017 рік.
Посада: студент кафедри безпеки інформаційних технологій з 2013 року.
Наукові інтереси: інформаційна безпека, програмування.
Публікації: матеріали та тези доповідей на наукових конференціях.
E-mail: taras1039@ukr.net

ПЕДЧЕНКО Євгеній Максимович

Рік і місце народження: 1998 рік, смт. Чорнухи, Полтавська обл., Україна.
Освіта: Національний авіаційний університет, студент.
Посада: старший лаборант кафедри безпеки інформаційних технологій з 2015 року.
Наукові інтереси: інформаційна безпека, персональні дані, програмування.
Публікації: матеріали та тези доповідей на наукових конференціях.
E-mail: zhenia1398@gmail.com

Анотація. *Постійний розвиток інформаційних систем впливає на всі сфери діяльності суспільства. Одним із актуальних напрямів, який активно розвивається у сфері інформаційної безпеки, є виявлення кібератак і запобігання вторгнень. Масовані кібератаки ініціюють створення спеціальних технічних рішень, засобів та систем протидії. Для виявлення мережевих вторгнень використовуються сучасні методи, моделі, засоби, програмне забезпечення і комплексні технічні рішення для систем виявлення та запобігання вторгнень, які можуть залишатись ефективними при появі нових або модифікованих видів кіберзагроз. На практиці при появі нових загроз та аномалій, зазначені засоби не завжди залишаються ефективними. Тому системи виявлення вторгнень повинні постійно досліджуватись і удосконалюватись. Серед таких систем є спеціалізовані програмні засоби, які направлені на виявлення підозрілої активності або втручання в інформаційну систему і прийняття адекватних заходів щодо запобігання кібератакам. Ці системи та засоби, як правило, достатньо дорогі, мають закритий код та вимагають періодичної підтримки розробників щодо їх удосконалення і відповідного налаштування до умов конкретних організацій. Враховуючи результати відомих досліджень в роботі проведений узагальнений аналіз програмних засобів систем виявлення вторгнень за визначеною базовою множиною характеристик («Клас кібератак», «Адаптивність», «Методи виявлення», «Управління системою», «Масштабованість», «Рівень спостереження», «Реакція на кібератаку», «Захищеність» та «Підтримка операційної системи»). Це надасть певні можливості для розробників і користувачів обрати відповідне сучасне програмне забезпечення для захисту інформаційних систем.*

Ключові слова: *атаки, кібератаки, аномалії, зловживання, системи виявлення вторгнень, системи виявлення кібератак, системи виявлення аномалій, виявлення аномалій в інформаційних системах.*

Інтенсивний розвиток інформаційних систем (ІС) та технологій всебічно впливає на всі сфери діяльності суспільства. Значна кількість сучасних державних та приватних підприємств використовує ІС для управління виробничими процесами, підтримки прийняття рішень, пошуку необхідних даних тощо. Це забезпечує їм низку переваг, пов'язаних з:

– підвищенням продуктивності праці і мобільності працівників;

– високою оперативністю доступу до інформації та послуг;

– можливостями віддаленого управління ресурсами і процесами тощо.

Разом з цим збільшується кількість уразливостей та загроз ІС і тому для забезпечення їх нормального функціонування та попередження вторгнень необхідні спеціалізовані засоби безпеки.

Слід зазначити, що одним із актуальних напрямів, який активно розвивається у сфері інформаційної безпеки є виявлення кібератак і запобігання вторгнень в ІС з боку неавторизованої сторони (НАС). Також слід наголосити, що атаки на ресурси ІС (РІС) з кожним роком стають все досконалішими, глобальнішими та частішими.

Масовані кібератаки [1-3] ініціюють створення спеціальних технічних рішень, засобів та систем протидії. Для виявлення мережевих вторгнень використовуються сучасні методи [4-12], моделі [12, 13], засоби [12, 14-16], программне забезпечення (ПЗ) [12, 17-27] і комплексні технічні рішення для систем виявлення та запобігання вторгнень [8, 12, 15, 22, 27-29], які можуть залишатись ефективними при появі нових або модифікованих видів кіберзагроз. Але на практиці при появі нових загроз та аномалій, породжених атакуючими діями з невстановленими або нечітко визначеними властивостями, зазначені засоби не завжди залишаються ефективними і вимагають тривалих часових ресурсів для їх відповідної адаптації. Тому системи виявлення вторгнень (СВВ) повинні постійно досліджуватись і удосконалюватись для забезпечення неперервності в їх ефективному функціонуванні.

Серед таких систем є спеціалізовані програмні засоби, які направлені на виявлення підозрілої активності або втручання в ІС і прийняття адекватних заходів щодо запобігання кібератакам. Ці системи та засоби, як правило, достатньо дорогі, мають закритий код та вимагають періодичної підтримки розробників (висококваліфікованих фахівців) щодо їх удосконалення і відповідного налаштування до умов конкретних організацій.

Виходячи з цього, проведення аналізу технічних рішень, спеціальних засобів та ПЗ виявлення кібератак, зловживань та аномалій в ІС для їх використання при виборі і розробці СВВ, а також визначення найбільш ефективних відповідних механізмів захисту РІС є актуальним завданням.

У [10, 20, 24, 26, 30] описано ПЗ AAFID, ASAX, NetSTAT, Prelude, Snort та SnortNet, яке використовується для виявлення порушень за такими характеристиками, як «Клас кібератаки», «Адаптивність», «Методи виявлення», «Управління системою», «Масштабованість», «Рівень спостереження», «Реакція на кібератаку», «Захищеність» та «Підтримка операційною системою (ОС)». Але для більш об'єктивної оцінки сучасного ПЗ важливо розглянути значно ширший спектр відповідних реалізацій, наприклад, OSSEC, Bro, Samhain, Suricata тощо.

Крім того, в [7, 8, 12, 15, 17, 21-23, 25, 26] наведений загальний опис окремих функції та принципи роботи ПЗ EMERALD, SIEM, OSSIM, CMDS, NetStat, Bro, Network Flight Recorder, Tripwire, Snort, Suricata, Prelude, NetProwler, NetRanger, Centrax та RealSecure, але не проведений аналіз відносно базових характеристик «Методи виявлення», «Реакція на кібератаку», «Захищеність» тощо.

Також в [27] розкриваються основні принципи функціонування найпопулярніших СВВ 2018 року – SolarWinds Log and Event Manager, Snort, OSSEC, Suricata, Bro, AIDE, OpenWIPS-NG, Samhain і Fail2Ban та визначені ОС, якими вони підтримуються, але не проведений аналіз відносно базових характеристик «Клас кібератак», «Адаптивність», «Захищеність» тощо.

У [18] порівнюються функціональності RealSecure, NetProwler, Snort та Форпост, в [24] здійснена оцінка Bro, RealSecure та Snort за функціональністю та продуктивністю, у [16] аналізуються OSSEC і Snort за можливостями моніторингу, видами сповіщень і попереджень про атаку та особливостями налаштування, а в [19] проведено аналіз СВВ з відкритим початковим кодом та порівняння конфігурації Snort 2, Snort 3 і Suricata 2 за низкою характеристик щодо кращого застосовування для захисту ІС. Але в цих роботах не проглядається узагальненість підходів та не проаналізовані сучасні засоби Bro, NetSTAT, Security Onion, Samhain тощо та не визначені їх властивості відносно базових характеристик «Реакція на кібератаку», «Клас кібератак», «Адаптивність», «Методи виявлення» тощо.

У роботах [6, 7, 15, 31] описана низка методів, які використовуються в ПЗ для виявлення атак і аномалій, але не проведена оцінка відносно характеристик «Масштабованість», «Рівень спостереження» та «Реакція на кібератаку».

В [16] порівнюються OSSEC і Snort за принципами налаштування, а в [32] розкривається склад СВВ та їх основні завдання, але не проведений аналіз відповідного найпоширенішого ПЗ за множиною характеристик «Захищеність», «Методи виявлення», «Масштабованість» тощо.

Також в [33] розглянуто системи виявлення та запобігання вторгнень, функціонування яких базується на аномаліях мережевого трафіку (аномальні системи виявлення та попередження вторгнень), в [12, 15] розкриваються методи та моделі, які використовуються для виявлення вторгнень, в [4-6, 8, 9, 11, 29, 34] порівнюються методи виявлення атак та аномалій, а в [29] акцентується увага на застосуванні нечіткої логіки для ефективного виявлення аномалій. Але в жодному з джерел не здійснено дослідження конкретного ПЗ, оцінки його властивостей та опису базових характеристик.

У [11, 12, 15, 17, 22, 29, 34] запропонована класифікація СВВ та систем запобігання вторгнень, зазначені їх переваги та недоліки і деякі особливості побудови, а в [7, 26] здійснена класифікація щодо виявлення мережевих вторгнень (аномалій і зловживань), але не розглядається існуюче ПЗ відносно визначених базових характеристик.

В роботах [6, 8, 10, 11, 13, 14, 18-20, 24, 26, 28, 29, 35, 36] розглянуті основні можливості, принципи побудови, механізми функціонування та порівняльний аналіз СВВ, але відсутне конкретне дослідження ПЗ щодо характеристик «Клас кібератак», «Адаптивність», «Методи виявлення» тощо.

У [15] проведений аналіз щодо проектування систем виявлення атак, показані основні принципи створення засобів протидії кібератакам, але відсутній аналіз відносно конкретного ПЗ щодо характеристик «Масштабованість», «Рівень спостереження», «Реакція на кібератаку» тощо.

Аналіз джерел [4-36] показав, що для сучасних ІС та мереж гостро стоїть питання оперативного виявлення зловживань та аномалій. В більшості зазначених робіт наведений лише частковий аналіз СВВ та їх класифікація, представлений загальний опис відповідного забезпечення, який не відображає їх широкого спектру та не містить необхідної множини характеристик для інтегрованої оцінки таких систем.

Виходячи з цього, метою роботи є проведення узагальненого аналізу програмних засобів СВВ за визначеною базовою множиною характеристик. Це надасть певні можливості щодо вибору таких засобів та розробки для них найбільш ефективних механізмів безпеки при впливах кібератак.

Як правило, методи виявлення атак розділяють на методи виявлення зловживань і аномалій [7, 30, 37, 38]. Зловживання засновані на використанні існуючих недоліків ІС. Основною відмінністю між аномалією і зловживанням є те, що аномалія це процес, який виникає перед можливим вторгненням в систему або вказує на наявність вже існуючої атаки. Фактично, аномалія – це відхилення від нормального стану системи, незвичайна активність в ній, що може свідчити про певні атакуючі дії. Слід зазначити, що аномалія може виникнути і за інших причин, наприклад, внаслідок неправильної роботи системи.

Саме тому за допомогою ефективного аналізу аномалій, що виникають у системі, можна попередити кібератаки певних типів і вчасно вжити необхідних заходів щодо їх блокування та захисту ІС.

Варто сказати, що широке використання сучасних засобів захисту від кібератак не гарантує безпеки на належному рівні, оскільки останнім часом:

– зростають атаки, спрямовані на корпоративні системи, публічні, конфіденційні та державні інформаційні ресурси;

– кібератаки, постійно модифікуються, удосконалюються і стають більш регулярними;

– виявлення кібератак класичними засобами захисту не завжди є ефективним;

– частішають випадки здійснення складних атак [1-3] на ІС [28, 39, 40].

Це також пов'язане з інтенсивним розвитком програмно-апаратних засобів і глобалізації інформаційних мереж та їх повсякденного використання у всіх сферах діяльності суспільства.

Враховуючи результати відомих досліджень з подальшим їх узагальненням і відображенням на розширений спектр засобів виявлення зловживань та аномалій проведемо аналіз сучасних СВВ відносно базових характеристик «Клас кібератак», «Адаптивність», «Методи виявлення», «Управління системою», «Масштабованість», «Рівень спостереження», «Реакція на кібератаку», «Захищеність» [30] та «Підтримка ОС» (див. таблицю 1).

Перед початком аналізу розкриємо кожну із зазначених базових характеристик.

«Клас кібератак» – визначає здатність системи виявляти аномалії та зловживання на різних рівнях ІС. Більшість сучасних засобів мають здатність виявляти обидва класи атак (аномалії та зловживання) [30].

«Адаптивність» – дозволяє системі ефективно адаптуватись до нових атак (відсутніх у базі даних сигнатур), наприклад, 0-day та виявляти кібератаки з незначними модифікаціями [30].

«Методи виявлення» – множини методів, що використовуються для виявлення атак і складають математичну основу системи. Найбільш поширеними є методи статистичного і кластерного аналізу, контролю зміни подій, графів атак, сигнатурні, динамічні, машинного навчання, поведінкові, евристичні, експертні, нечітких множин тощо [7, 30, 31, 38, 41].

«Управління системою» – визначає схему управління і його рівень. Управління може здійснюватися централізовано із одного хоста або розподілленно із окремих хостів, пов'язаних однією системою. Найбільш оптимальною є організація управління за централізованою схемою з певною множиною центрів, кожний з яких може бути задіяний для управління всією структурою [30]. Централізовані системи реалізують управління всіма засобами (модулями) виявлення аномалій та зловживань з однієї станції [39], а розподілені реалізують управління окремо, де кожний модуль відповідає за свою функцію [42].

«Масштабованість» – можливість розширення системи, її адаптивність до різних мережевих структур та долучення нових аналізованих ресурсів мережі [30].

«Рівень спостереження» – визначає, на якому рівні системи отримуються дані для виявлення кібератак. Застосовуються два рівні отримання даних – мережевий та системний. Сучасні системи, як правило, підтримують обидва рівні спостереження, оскільки саме їх взаємодія дозволяє краще забезпечити захист. Від цієї характеристики залежить швидкість формування первинних даних, їх правильна обробка та отримання точної інформації про поточний стан РІС [30].

Аналіз трафіку мережі здійснюється за допомогою спеціальних сенсорів (мережевих і системних), що застосовуються у системах виявлення атак та аномалій. Мережеві сенсори аналізують дані на мережевому рівні (зазвичай на основі сигнатурного аналізу) і генерують повідомлення про виявлення кібератак та відправляють їх до модулів управління.

Системні сенсори аналізують журнали реєстрації ОС, додатки та програмні застосунки на можливі аномалії чи загрози і генерують відповідні повідомлення, які надходять до модулів управління [30].

«Реакція на кібератаку» – визначає наявність у системі компонентів чи модулів протидії. Тобто після реєстрації атаки ініціюються дії для редукування подальшого негативного впливу [30].

«Захищеність» – характеризує наявність власних компонентів системи, які відповідають за її захист від кібератак та зовнішнього негативного інформаційного впливу, а також за стійкість до виходу з ладу та зменшення кількості уразливостей розробки в цілому [30].

«Підтримка ОС» – характеризує тип ОС (наприклад, Unix, Linux, Windows, MacOS тощо), що підтримує відповідне ПЗ системи.

Далі з урахуванням запропонованих характеристик розкриємо властивості відповідного СВВ (див. таблицю 1).

AAFID

Система AAFID (Autonomous Agents for Intrusion Detection, розробка Purdue University, West Lafayette, Індіана, США) призначена для розподіленого контролю та виявлення вторгнень. Використовує невеликі автономні програми (агенти) для виконання функцій моніторингу в хостах мережі. Архітектура AAFID засновується на незалежних одночасно працюючих об'єктах (агентах), які направлені на виявлення вторгнень. Вони контролюють визначену множину характеристик системи та повідомляють про нестандартну поведінку або конкретні події. Інформація, що отримана агентами, інтегрується на рівні головного комп'ютера, де здійснюється співвідношення подій (отриманих від різних агентів), які можуть бути викликані однією і тією ж атакою. Крім того, звіти, що надаються з кожного комп'ютера агрегуються на більш високому рівні (рівень мережі), що дозволяє системі виявляти кібератаки з різних джерел (рис. 1-2) [43].

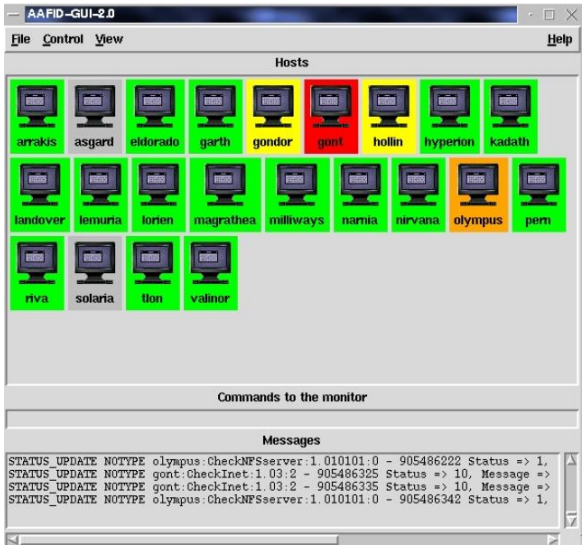

Рис. 1. Головне вікно прототипу ПЗ AAFID

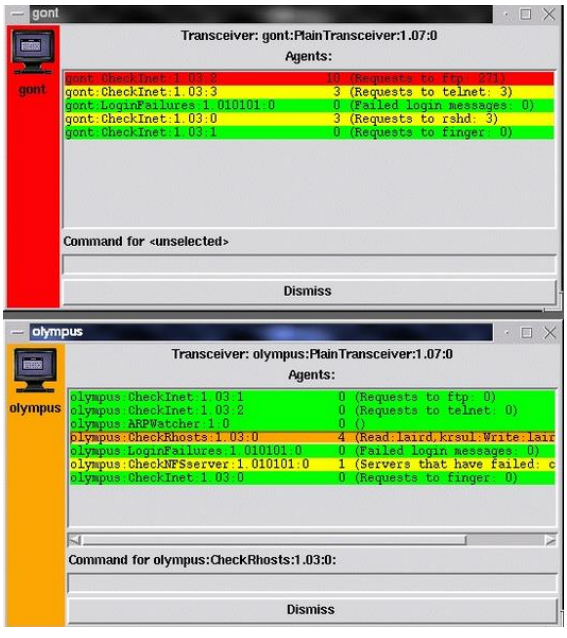

Рис. 2. Вікно агента прототипу ПЗ AAFID

Даний програмний засіб здійснює обробку тільки журналів реєстрації програмних застосунків та ОС, в якій він функціонує. Всі агенти працюють за сигнатурним принципом, вони виявляють відомі (заздалегідь описані) аномалії, зловживання та події на вузлах і в мережі. Система знаходиться на стадії прототипу. Множина агентів, що входить до AAFID у явному вигляді використовує характеристики атак, сформовані експертами (розробниками агентів). При цьому не використовується ніякої формальної мови опису кібератак. Агенти є програмними модулями, написаними на алгоритмічній мові загального призначення RUSSEL, в яких жорстко визначені ознаки тих атак, на виявлення яких вони орієнтовані. Така організація бази описів відомих атак (NADF) не є гнучкою щодо розширення і відповідно не є адаптивною до нових кібератак [38]. Для збирання інформації з кожного агента або хоста (набора хостів) в AAFID використовується ієрархічна структура. Таким чином виявляється будь-яка підозріла активність в мережі, хоча зазначена властивість не завжди працює ефективно, оскільки можливості системи залежать від якості і кількості наявних агентів (програм), які використовуються для виявлення тих чи інших атак [44].

Зазначений засіб використовує експертний та сигнатурний метод виявлення аномалій і зловживань в мережевому трафіку та має централізоване управління з основного монітора [30]. Завдяки тому, що система має ієрархічну структуру, то AAFID можна легко модифікувати. Система має відкритий інтерфейс для додавання нових агентів і фільтрів, що дозволяє їй просто масштабуватись і адаптуватись під потреби мережі [45, 46].

Варто зауважити, що AAFID сама по собі не є мережевою СВВ. Частина агентів реалізують функції моніторингу мережі, а інші виконують функції моніторингу хоста. У цій архітектурі вузли СВВ розташовуються відповідно до деревовидної ієрархічної структури [44].

Слід зазначити, що AAFID не містить спеціальних механізмів захисту та реакції на вторгнення і не є стійкою до можливих кібератак, які на неї спрямовані [30, 44-46]. Вона підтримується такими ОС як Unix та Linux [44].

Snort

Snort [47] (розробка компанії Sourcefire, США) на світовому рівні є найпоширенішою безкоштовною мережевою системою виявлення та запобігання вторгнень (рис. 3-4) [8, 12, 27, 48].

Структурно Snort підтримує декілька режимів функціонування:

– аналіз пакетів;

– журналювання (протоколювання) пакетів;

– виявлення мережевих вторгнень;

– інші вбудовані можливості [49].

Архітектура системи розроблена з урахуванням ефективності та швидкості в роботі. Тому вона абсолютно проста і складається з:

– декодера пакетів;

– ядра виявлення;

– підсистеми оповіщення та реагування [30, 49].

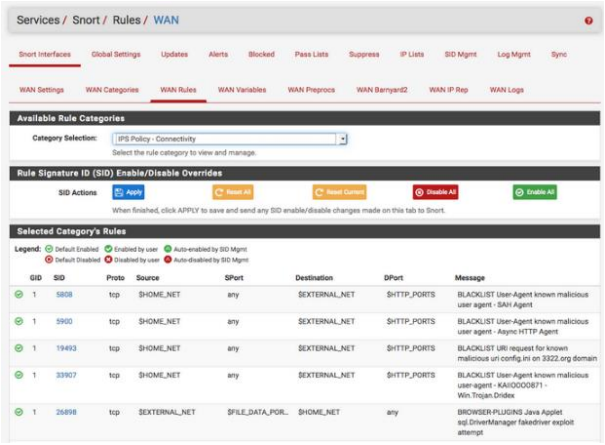

Рис. 3. Вікно відображення налаштування правил Snort

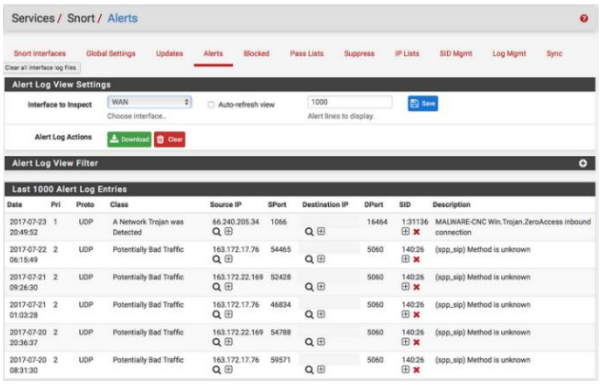

Рис. 4. Вікно відображення сповіщень, згенерованих Snort

Декодер реалізує набір процедур для послідовної декомпозиції пакетів відповідно до рівнів мережевого стека, тобто прийнятий кадр послідовно перетворюється в пакет, сегмент і блок даних з урахуванням специфічних для даного рівня атрибутів сигнатур. Підтримуються протоколи канального рівня Ethernet, SLIP, PPP та ATM [30].

Ядро інтегрує існуючі правила в ланцюги, які складають відповідні двомірні послідовності, за якими здійснюється проходження кожного пакету [30].

Підсистема оповіщення та реагування відповідає за збереження результатів аналізу трафіку в журналах реєстрації Snort або передачу цієї інформації системними службами реєстрації подій ОС [30, 49].

В системі використовується проста мова опису атак, яка повністю є в документації і дозволяє адміністраторам самостійно розширювати базу сигнатур. Кожне правило складається з двох частин – умови його застосування та дії. Крім того, в останніх версіях системи з'явилася спеціальна конструкція мови сигнатур, що дозволяє класифікувати мережевий трафік за ступенем потенційної небезпеки, який визначається експертом, що формує атрибути кібератаки. Також Snort виконує функції протоколювання, аналізу і пошуку за вмістом та широко використовується для активного блокування або пасивного виявлення цілої низки зловживань і аномалій (використовуються засоби інспекції протоколів і механізми виявлення аномалій), наприклад, пов'язаних з атаками на пере-

повнення буфера, прихованим сканування портів, атаками на веб-додатки, SMB-зондуванням, спробами визначення ОС тощо. Відповідне ПЗ в основному використовується для запобігання проникнення та блокування поточних кібератак [50].

Система функціонує на основі сигнатурного методу. Це дозволяє швидко виявляти всі задекларовані нею кібератаки. Але через неможливість повноцінного виявлення нових атак у мережі вона не є повністю адаптивною. Система Snort є програмним продуктом з відкритим вихідним кодом, що дозволяє легко змінювати її структуру. Вона здатна виконувати реєстрацію пакетів і в режимі реального часу здійснювати аналіз трафіку в IP-мережах. Також завдяки відкритій архітектурі та відкритому початковому тексту система стала швидко розвиватися (за рахунок інших розробників) і інтегруватися з різними програмними продуктами, наприклад, такими як бази даних журналів виявлення, аналізатори журналів реєстрації тощо [51].

Модуль аналізу трафіку базується на основі правил (сигнатур). До ядра виявлення можуть інтегруватися модулі сторонніх розробників (препроцесори) і проводити аналіз на одному з рівнів декомпозиції пакетів. За допомогою таких модулів можна розширити функціональність ядра виявлення та реалізовувати різні методи виявлення. Також до складу Snort був доданий модуль статистичного аналізу, який призначений для виявлення аномалій в мережевому трафіку [8].

У системі реалізоване централізоване управління за допомогою однієї станції. Оскільки Snort є представником системи з відкритим кодом, то продукт легко масштабувати та змінювати під власні потреби. Система дозволяє ефективно використовувати існуючі та самостійно створювати нові правила для виявлення атак виключно на основі аналізу мережевого трафіку. Підсистема оповіщення та реагування включає базові методи реакції на кібератаку – розрив з'єднання з атакуючим об'єктом чи блокування його. Механізми захисту в Snort реалізуються протоколом SNMPv2, у якому застосовуються функції шифрування паролів при передачі даних [30, 49, 51]. Програмний засіб Snort працює на ОС Unix, Linux та Windows [47].

Prelude SIEM

Універсальна система Prelude SIEM (Security Information & Event Management – управління інформацією про безпеку, розробка США) (рис. 5-6) збирає, нормалізує, сортує, корелює та звітує про всі події, пов'язані з безпекою незалежно від того, що породжує ці події. Також Prelude користується підтримкою інших подібних систем (snort, samhain, ossec, auditd тощо), що дозволяє покращити її функціонування [8, 12, 52].

Зазначене ПЗ є розподіленою гібридною СВА, яка складається з наступних базових компонент:

– ядро;
– агент;
– модуль кореляції;
– база даних;
– підсистема обміну повідомленнями;
– основний інтерфейс;
– модуль управління [53].

Рис. 5. Головне вікно Prelude SIEM

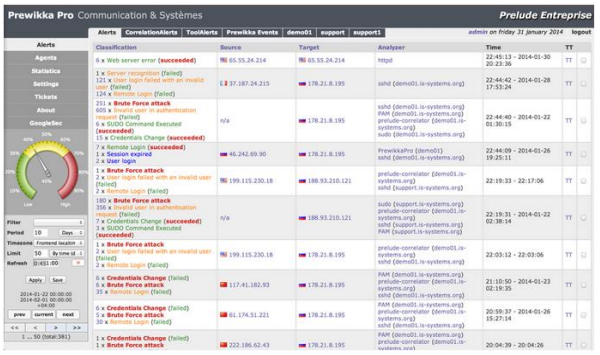

Рис. 6. Основний інтерфейс Prelude SIEM

Ядро системи відповідає за прийом нормалізованих подій (від агентів, модулів кореляції, сторонніх систем або підпорядкованих менеджерів), запис у базу даних та інформування через e-mail. Агент системи працює локально (на одному сервері) та віддалено і здійснює прийом логів від різних систем (через локальний файл або syslog на UDP-порт), розбирає або нормалізує їх на основі множини правил, що складаються з регулярних виразів, а нормалізовані події направляє ядру. Модуль кореляції підключається до ядра як агент і корелює події, що надійшли до ядра на основі плагінів, реалізованих у вигляді Python-скриптів. База даних зберігає всі події, які обробляються системою. Підсистема обміну повідомленнями включає додаткові ресурси, що безпосередньо підключаються до ядра за підтримки IDMEF (Intrusion Detection Message Exchange Format – спеціальний формат обміну повідомленнями про вторгнення). Основний інтерфейс реалізований на протоколі http і призначений для відображення результатів обробки подій, їх агрегації або фільтрації, виведення статистичної інформації тощо [53].

Система включає в себе модуль управління, який отримує і обробляє повідомлення сенсорів та генерує можливу реакцію на атаку, наприклад, блокування порушника на мережевому екрані (Net/IP Filter). Агенти реагування (відповідно до згенерованої реакції) реалізують необхідні заходи протидії кібератакам. Додаткові модулі аналізу мережевих даних роблять систему стійкою до некоректних мережевих пакетів на різних рівнях стека та виходу її компонентів з ладу. Це пов'язано з відправкою пакетів з неправильними контрольними сумами, синхронізацією сесій, випадковими відправленнями та іншими діями, що ігноруються [30].

Система побудована на сенсорах мережевого та вузлового рівнів. Перші аналізують вхідні данні на рівні мережі та генерують повідомлення щодо виявлення атак і відправляють їх модулям управління. Другі аналізують журнали реєстрації ОС та програмних застосунків (сенсори рівня системи), генерують повідомлення про виявлення аномалій і відправляють їх модулям управління. Мережеві сенсори орієнтовані на виявлення зловживань у системі, а вузлові на виявлення аномалій. Prelude заснована на сигнатурному підході, що дозволяє швидко виявляти всі задекларовані у системі атаки. Застосований підхід не ефективний відносно нових загроз, які не відображені у базі даних і тому система не є адаптивною до нових кібератак. Зазначена розробка є системою з відкритим початковим кодом, що дозволяє її помодульно реконфігурувати. Вона, в основному, використовує метод протоколювання подій та шаблони атак. Управління здійснюється централізовано за допомогою керуючої консолі, якій компоненти системи самі надають ті параметри щодо їх функціонування, які можуть змінюватися. Також управління може здійснюватися через локальні конфігураційні файли на тих вузлах, де встановлені компоненти системи. Вся архітектура відкритої системи Prelude побудована за принципом використання відкритих стандартів. Підсистема обміну повідомленнями IDMEF дозволяє легко масштабувати та адаптувати зазначену розробку під різні потреби, а також інтегрувати її компоненти в системи сторонніх виробників і навпаки. Архітектура Prelude дозволяє адміністратору мережі стежити за активністю на рівні мережі та на рівні окремих вузлів. При розробці системи особливу увагу було приділено питанням безпеки та захищеності. Канали передачі даних шифруються за протоколом SSL, а також використовується спеціалізована бібліотека, яка запобігає класичним помилкам виходу за межі масивів і переповнення буферів [30]. Програмний засіб Prelude працює на ОС Linux [52].

NetSTAT

В основі системи NetSTAT (Network-based State Transition Analysis Tool, розробник кафедра комп'ютерних наук університету Каліфорнії, Санта-Барбара, США) закладена розширювана мова опису атак та їх шаблонів (STATL). Базова мова використовує найбільш абстрактні поняття і не залежить від конкретної системи та її конфігурації. Мова дозволяє самостійно добудовувати себе, є розширюваною, а додаючи специфічні для конкретної системи події, може бути легко адаптована до різних цільових середовищ. Для кожної нової події описується проникнення у вигляді послідовності дій. Такий опис групується в модуль розширення мов і його можна використовувати в описі сценаріїв кібератак для NetSTAT [23, 54].

Система має два режими функціонування. Перший заснований на тому, що для кожного стану визначається характеристика захищеності та переходи при зміні стану системи. Атаки описуються у вигляді послідовних переходів. Другий ґрунтується на сигнатурному підході, тобто описі атак у вигляді послідовності переходів та шаблонів, з якими здійснюється порівняння NetSTAT для виявлення вторгнень в мережевому середовищі орієнтована на функціонування в режимі реального часу [55].

Основною функціональною частиною системи є ядро, яке експлуатує абстрактні об'єкти та події і не залежить від конкретної системи. Тут здійснюється порівняння вхідного потоку з наявними сценаріями атак чим фактично реалізується функція виявлення. Для генерації потоку подій використовується джерело подій. Це програмний компонент системи виявлення, що здійснює перетворення інформації з системних джерел (наприклад, журналів реєстрації) у придатний для обробки формат. У системі також можуть використовуватися модулі протидії, що пов'язані з ядром і реалізують реакцію на виявлену кібератаку [30], а при передачі даних щодо стану NetSTAT здійснюється шифрування за протоколом SSL [30, 54-56].

Особливості будови системи дають їй можливість виявляти зловживання та аномалії у мережі. Наявність двох принципів роботи дозволяє ефективно знаходити і виявляти кібератаки певного типу. Використання методу реєстрації переходів станів частково дає можливість ідентифікування нової аномалії чи атаки, але не вирішує в повній мірі проблему адаптивності системи. Дана розробка відкрита і дозволяє будувати масштабовані СВВ виходячи із задач організації. В основу функціонування NetSTAT закладено метод, що описує підзахисну систему у вигляді набору станів її компонентів з подальшим аналізом їх переходів у результаті активних зовнішніх впливів. За необхідністю для фіксації переходів створюються журнали реєстрації подій для подальшого полегшення виявлення аномалій чи зловживань у системі. Управління здійснюється розподілено, оскільки система має розгалужену і складну структуру. Архітектура NetSTAT дозволяє будувати агенти або сенсори, що призначені для виявлення атак чи зловживань на різних рівнях мережі чи системи (рис. 7) [30, 54-56]. Спостереження за системою виконується на системному та мережевому рівнях. Також NetSTAT може визначити точки та мережеві події, які необхідно контролювати [55].

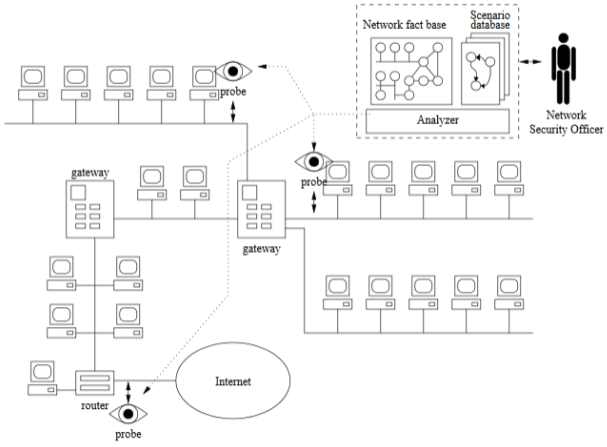

Рис. 7. Архітектура системи NetSTAT

Програмний засіб NetSTAT працює на ОС Unix, Linux та Windows.

ASAX

ASAX (Advanced Security audit trail Analyzer on uniX, спільна розробка Університету Namur та Siemens Nixdorf Software S.A., Бельгія) є універсальною експертною системою для виявлення вторгнень. Її принцип роботи полягає в описі початкової системи у вигляді множини станів з їх подальшим аналізом. Для ASAX розроблена проста мова RUSSEL, яка використовує спеціальні правила (рис. 8) для ефективної обробки великих послідовних файлів, що базуються на аналізах журналів реєстрації. RUSSEL можна розглядати як процедурну мову, включаючи конкретну, заздалегідь визначену структуру управління, яка підходить для обґрунтування послідовностей записів. Ця структура управління базується на певному механізмі, що запускає правило, до якого входить опис умови його спрацювання та наступні дії (наприклад, висновок, повідомлення або виклик іншого правила). Головне правило є основою всієї множини, яке активується першим і далі викликаються ті, що знаходяться в полі його дії. Також можливо використовувати систему для обробки даних журналів реєстрації в режимі реального часу [57-58].

```
rule criticalFileAccess;
begin
if
  ret_err = 0              /* success   */
  and (event = 4           /* creat(2)  */
    or event = 6           /* unlink(2) */
    or event = 10          /* chmod(2)  */
    or event = 39          /* fchmod(2) */
    or event = 42          /* rename(2) */
    or (73 <= event <= 83) /* open(2)   */
  and (isPrefix(fname, '/etc/aliases')
    or isPrefix(fname, '/etc/group')
    or isPrefix(fname, '/etc/passwd')
    or isPrefix(fname,
                '/var/spool/cron/crontabs')
    or isStartUp(fname))
  -->
  updt_fact_base(event, fname, uid, gid, mode)
fi;
trigger off for_next criticalFileAccess
end;

init_action;
begin
    trigger off for_next criticalFileAccess;
    init_fact_base('datalog.log')
end.
```

Рис. 8. Вікно ASAX з прикладом правила виявлення доступу до критичних файлів

ASAX орієнтована на виявлення тільки зловживань на рівні ОС, але проста реалізація експертного методу не дає можливість виявляти нові атаки на мережу та адаптовуватись до нових кібератак. В основу ASAX закладений простий варіант реалізації експертних СВВ. Вона має відкритий інтерфейс і програмний код, що дозволяє розширяти функціональні можливості. Даний програмний засіб має централізоване управляється на вузлі його встановлення за допомогою файлів конфігурації. Простота масштабування зумовлюється простою структурою побудови ASAX. Оскільки система працює тільки з журналами реєстрації додатків і ОС, то для неї характерний системний рівень спостереження. [30, 57-59].

ASAX не містить спеціальних механізмів захисту та реакції на кібератаки і не є стійкою до реалізації можливих загроз, які спрямовані на неї, працює на ОС Unix і Linux [60].

Bro

Система Bro (спільна розробка національної лабораторії Лоуренса та центру дослідження Інтернету ICSI в міжнародному інституті комп'ютерних наук, Каліфорнійський університет, Берклі, США) є автономним ПЗ для виявлення мережевих вторгнень у режимі реального часу шляхом пасивного контролю за мережевими посиланнями [8, 12, 23, 27]. Дана система (рис. 9-10) є відкритою, безкоштовною і розповсюджується за власною відкритою ліцензією та працює під управлінням декількох варіантів ОС UNIX. Bro є ключовою частиною інфраструктури безпеки центру суперкомп'ютерних застосунків NCSA. Її платформа надає широкий спектр можливостей щодо аналізу трафіку, який охоплює заголовки пакетів, регулярні вирази, фіксацію станів високорівневих з'єднань, статистичний аналіз тощо [61].

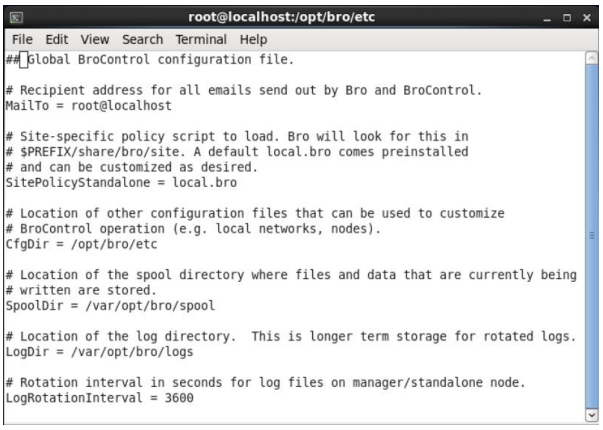

Рис. 9. Конфігураційний файл Bro

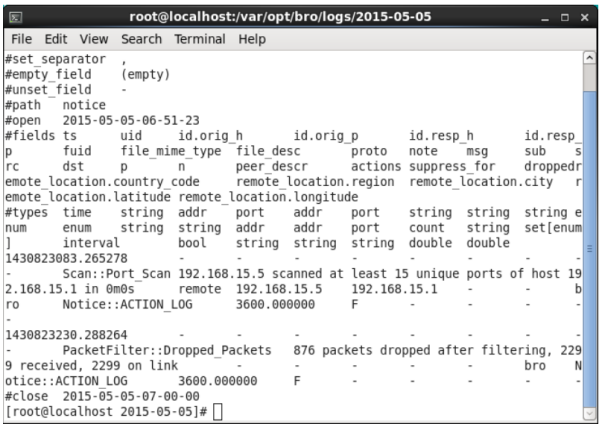

Рис. 10. Лог-файл з повідомленням сканування

Основний функціонал системи направлений на:

– високошвидкісний моніторинг великого обсягу інформації;

– контроль перевантажень;

– механізм поділу;

– масштабованість;

– здатність протистояти кібератакам;

– інформування в режимі реального часу [62].

Функція виявлення вторгнень в Bro пов'язана з етапами реєстрації та аналізу трафіку. Модуль

аналізу системи має два елементи, які орієнтовані на аналіз сигнатур та виявлення аномалій [27].

Система має ієрархічну архітектуру з трьома рівнями функцій. На першому рівні використовується утиліта libpcap для вилучення з мережі пакетів з даними. На другому виконується перевірка цілісності пакетів за заголовками, а при виявленні помилок генерується відповідне повідомлення. На третьому рівні згенеровані події розміщуються в черзі, яка опрацьовується інтерпретатором сценарію політики на основі мови Bro. Зазначена множина сценаріїв є політикою безпеки мережі, яка визначає реакцію системи на різні події. Сценарій може генерувати повідомлення, а також виконувати будь-які команди ОС і фактично реагувати на атаки. Сценарії політики можна налаштувати, але зазвичай вони працюють за стандартною схемою, яка включає підписування, виявлення аномалій та аналіз з'єднань. Виконання коду може закінчитися генерацією подальших подій, реєстрацією повідомлення в реальному режимі часу або протоколюванням. Щоб додати нову функцію до можливостей, Bro необхідно підготувати відповідний опис ідентифікації подій і обробник подій. На даний момент Bro контролює чотири прикладних сервіса – finger, ftp, portmapper і telnet [27, 62-64].

При виявленні атакуючих дій, система може повідомити оператора про підозрілу активність, записати повідомлення у відповідний журнал або виконати команди (правила) занесені до системи. Систему можна встановити на Unix, Linux або MacOS для виявлення зловживань та аномалій у мережі. Наявність модуля контроля перевантаження дає можливість обробляти великі обсяги даних без зниження пропускної здатності мережі. Якщо НАС спробує перевантажити мережу сторонніми пакетами для виведення СВВ з ладу, то Bro буде змушена пропускати пакети, серед яких можуть виявитися ті, що створені зловмисником для проникнення в мережу. Наявність такого модуля не вирішує проблему адаптивності системи. Розмежування процесів фільтрації даних, ідентифікації подій і політики реагування на них спрощує експлуатацію і обслуговування системи. Вона має простий механізм внесення додаткових записів про нові типи нападів і фактично заснована на сигнатурному підході виявлення атак. Дана система управляється централізовано та є зручною в управлінні завдяки певним командам взаємодії. Для виявлення нових уразливих місць, а також захисту від відомих типів загроз існує можливість швидкого додавання нових сценаріїв нападу у відповідну внутрішню бібліотеку. Bro використовують для пасивного моніторингу мережевого трафіку та пошуку підозрілої активності і аномалій в мережі. Система реалізує виявлення на прикладному рівні (використовуються ситуаційно-орієнтовані аналізатори для порівняння з шаблонами атак) та аналізує вхідний мережевий трафік (виявлення на семантичному рівні програмних застосунків). Слід зазначити, що складні сценарії нападу неодмінно включають елементи впливу на СВВ, але в системі відсутні засоби захисту для каналів передачі даних [62, 63]. Bro підтримується ОС Unix, Linux та MacOS [65].

OSSEC

Система OSSEC (Open Source SECurity, розробка Daniel B., корпорація Atomicorp є виробником ОС Linux, яка включає OSSEC як одну з основних технологій, США) це масштабована, багатоплатформена, вузлова СВВ на основі хоста з відкритим вхідним кодом (рис. 11) [12, 27, 66]. Має потужний інструмент кореляції та інтегрованого аналізу журналів, перевірки цілісності файлів, моніторингу реєстру Windows, централізованого нагляду за політикою, виявлення руткітів, оповіщення про атаки в режимі реального часу, виявлення закладок та протидії на вторгнення. Вона працює на ОС Linux, OpenBSD, FreeBSD, MacOS, Solaris, Windows та іншими [67, 68].

При виникненні вторгнень завдяки відповідним журналам, що надсилаються на електронну пошту можна дізнатись про атакуючі дії та вжити необхідних заходів. Також OSSEC може експортувати попередження в будь-яку систему SIEM за допомогою системного журналу. Це дає можливість користувачам отримувати аналітичні матеріали в реальному режимі часу та проглядати і аналізувати події в системі безпеки [67].

Система використовує агенти (сукупність невеликих програм, встановлених на систему для моніторингу), які збирають інформацію та передають її менеджеру для аналізу та кореляції (рис. 12). Частина інформації збирається в режимі реального часу, а частина з певним періодом. OSSEC може бути встановлена на Microsoft Windows платформу і виконувати функції агента [69]. Система використовується інтернет-провайдерами, університетами, урядами і великими корпоративними центрами обробки даних [67, 70].

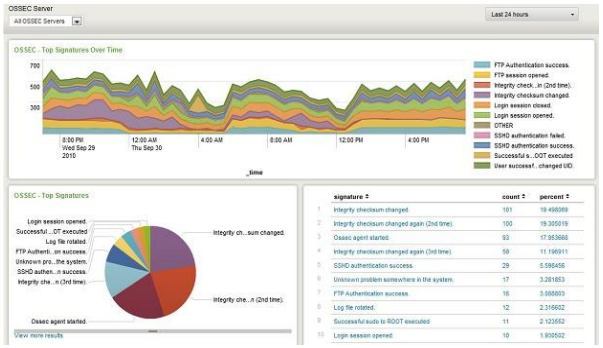

Рис. 11. Вікно відображення сервера OSSEC

Також OSSEC містить низку аналізаторів виявлення загроз для різних джерел даних, реалізує функції контролю цілісності файлової системи, виявлення сигнатур відомих троянських закладок (rootkits) тощо. З урахуванням можливості контролю цілісності ресурсів у вузлах можна говорити про умовну адаптивність OSSEC. Система повністю вільна у використанні і має відкритий інтерфейс для додавання нових модулів аналізу. Її можна адаптувати до своїх потреб безпеки завдяки широким можливостям налаштування, формуючи власні правила сповіщення та написання скриптів, які вживають

заходів у відповідь на порушення безпеки. OSSEC може змінювати початковий код для розширення функціональних можливостей. Система використовує сигнатурні методи виявлення кібератак і може бути встановлена в одиночній конфігурації на одному вузлі або в розподіленій на декількох вузлах (в такому випадку одна з інсталяцій стає сервером, а решта є агентами системи). При цьому управління агентами здійснюється централізовано з сервера. OSSEC на вузлах, де встановлені агенти управляється розподілено (за допомогою файлів конфігурації) або централізовано за допомогою спеціалізованої утиліти адміністрування (Manage_agents) з центрального сервера. Завдяки цьому система є добре масштабованою. OSSEC працює виключно з журналами реєстрації додатків і ОС та дозволяє використовувати довільні команди для реагування на атаку. Для цього необхідно статично задати відповідність між подією, командою і параметрами її виклику. При передачі інформації про поточний стан системи здійснюється шифрування за протоколом SSL [30, 67, 70].

Рис. 12. Вікно OSSEC Agent Manager
для введення IP-адреси менеджера

Suricata

Програмний засіб Suricata (розробка компанії Open Information Security Foundation, Бостон, США) має відкритий код, є безкоштовним, швидким, надійним та перспективним засобом виявлення мережевих загроз (рис. 13). Він призначений для запобігання та виявлення вторгнень у режимі реального часу, моніторингу мережевої безпеки, автоматичного аналізу та обробки PCAP-файлів [8, 27, 71].

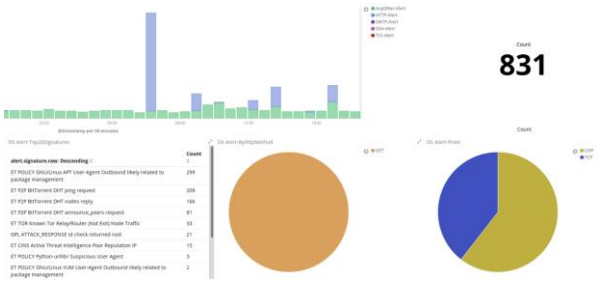

Рис. 13. Функціональне вікно програмного засобу Suricata

Suricata працює на рівні додатків і це дозволяє виявляти загрози, які можуть залишатися непоміченими. Контроль здійснюється на рівні протоколів TLS, ICMP, TCP, UDP, HTTP, FTP та SMB, а також є можливість виявляти спроби вторгнень, що приховуються під звичайними запитами та існує функція вилучення файлів для їх перевірки. Архітектура Suricata дозволяє оптимально розподілити обчислювальне навантаження між декількома ядрами процесора. Наприклад, якщо відеоадаптери більшість часу знаходяться в неактивному режимі, то їх частково можна завантажити певними обчисленнями [72].

Також програмний засіб здатний виявляти уразливості в режимі реального часу, попереджувати вторгнення в систему, переглядати властивості мережевої безпеки [71] та поєднувати властивості виявлення аномалій і зловживань [73]. Крім того, Suricata має здатність адаптуватись до нових атак, працювати з іншим ПЗ (наприклад, Splunk, SIEM, Kibana тощо), контролювати мережевий трафік (використовуючи сигнатури та розширені правила подібні до Snort) і має потужну підтримку сценаріїв Lua для виявлення складних загроз [71].

Наявні засоби перевірки HTTP-трафіку засновуються на бібліотеці HTP. Також здійснюється контроль файлів (що передаються з використанням HTTP), розбір стисненого контенту, ідентифікація за URI, cookie, заголовками тощо. Контент в потоці можна виділяти за допомогою маски і регулярних виразів, а ідентифікація файлів можлива за іменем, типом або контрольною MD5-сумою [74]. Програмний засіб має централізоване управління [72] і швидко виявляє уразливості та атаки завдяки розподіленій роботі між ядрами процесора та потоками [73]. Спостереження за системою відбувається на системному і мережевому рівнях [27].

В Suricata реакція на кібератаку здійснюється оперативно у тому випадку, якщо порушено не меньше одного із налаштованих правил, шляхом маркування отриманих пакетів даних, одним із трьох маркерів:

– NF_ACCESS (доступ наданий);
– NF_DROP (доступ заборонений);
– NF_REPEAT (пакети маркуються та повторно направляються на правила брандмауера, який і вирішує подальше призначення відповідного пакету) [73].

Даний програмний засіб є загальнодоступним для всіх користувачів і він не має механізмів захисту [71]. Suricata функціонує на ОС Unix, Linux, Windows та MacOS [27].

Samhain

Система Samhain (розробка компанії Samhain Services, Люнебург, Німеччина) є відкритим, безкоштовним, мультиплатформеним ПЗ для СВВ [27, 72, 75]. Її також називають хост-системою, що забезпечує перевірку файлів, перегляд та аналіз логів, виявлення зловмисного коду (в SUID файлах), прихованих програм та процесів тощо [72].

Samhain (рис. 14) розроблена, як монітор для багатьох хостів з різними ОС та для локальних комп'ютерів [75]. Однією з її функцій є стелс-режим, який дозволяє маскуватися від НАС. Цей режим

використовує стеганографію для приховування своїх процесів від інших. Також для запобігання вторгнень Samhain захищає свої центральні файли журналів та резервні копії конфігурацій за допомогою PGP [72].

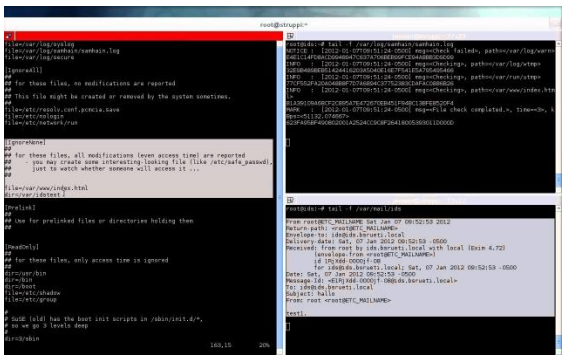

Рис. 14. Вікно роботи програмного застосунку Samhain

Програмний засіб здатний працювати в реальному режимі часу, здійснювати перевірку файлів та логів системи, виявляти приховані програмні засоби, шкідливе ПЗ і аномалії [76].

Samhain завантажується як демон системи (служба) та приховано здійснює виявлення загроз. Стелс-режим дозволяє системі бути невидимою для шкідливого ПЗ і тому НАС не буде намагатися протидіяти відповідному монітору, якщо попередньо не буде знати про нього. Завдяки такому режиму вона здатна зупиняти чи перезавантажувати процеси, що дає змогу виявити можливі загрози. Зазначене ПЗ призначене більше для серверного використання і здатне здійснювати перевірку з'єднання із сервером та ідентифікацію правильності введеного паролю при вході в систему. Цілісність звітів, які формуються Samhain забезпечується алгоритмом AES, що ускладнює їх модифікацію шкідливим ПЗ [77].

Протидія атакам в Samhain здійснюється на трьох рівнях:

– перший базується на перевірці контрольної суми (використовуються криптографічні контрольні суми файлів для виявлення модифікацій та шкідливого коду в SUID файлах, розташованих на диску);

– другий засновується на централізованому моніторингу (існує вбудована підтримка входу до центрального серверу шляхом шифрування та автентифікації підключень);

– третій забезпечує захищеність від зламування (баз даних, конфігураційних та лог-файлів, e-mail звітів що можуть підтримувати приховані операції тощо) [77].

В системі підтримується централізоване і розподілене управління [76], а також є можливість адаптування та масштабування відповідно до кількості хостів (у випадку активації даного ПЗ на серверній машині), на яких здійснюється попередження та виявлення активності НАС [77].

Спостереження за такою активністю в Samhain відбувається тільки на системному рівні [27], а реакція на кібератаку відбувається в реальному режимі часу [75]. Крім того, Samhain підтримується ОС Unix, Linux, MacOS і Windows 2000/XP (завдяки емулятору Cygwin) [77] та використовує механізми захисту, які не розкриваються виробником [75].

Security Onion

Система Security Onion (розробка компанії Security Onion Solutions, США) є безкоштовним і відкритим ПЗ для ОС Linux, яке направлене на виявлення вторгнень, моніторинг стану безпеки підприємств, управління і перегляд системних журналів. Воно містить простий у використанні майстер налаштування розподілених давачів (рис. 15-17) та інтегрує відомі засоби безпеки Elasticsearch, Logstash, Kibana, Snort, Suricata, Bro, Wazuh, Sguil, Squert, CyberChef, NetworkMiner тощо [27, 78].

Для ефективного функціонування Security Onion потребує:

– попереджувальні дані (формуються за результатами локального спостереження за допомогою Wazuh та мережевого за допомогою Snort або Suricata);

– дані про активи (спостереження за активами підприємства здійснює Bro);

– повний вміст даних (повний перегляд пакетів даних, що циркулюють, здійснюється завдяки netshif-ng);

– локальні дані (спостереження за локальними даними здійснюється за допомогою Beats, Wazuh, syslog тощо);

– дані сесії (перегляд сесійних даних відбувається за допомогою Bro);

– дані про транзакцій (дані, що надіслані через http/ftp/dns/ssl переглядаються за участю Bro) [79].

Рис. 15. Вікно інсталяції Security Onion

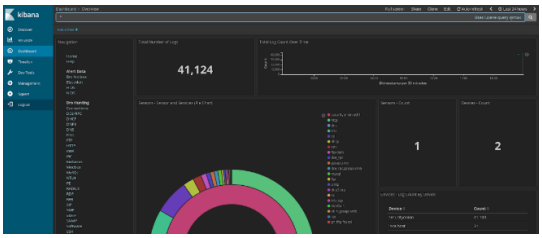

Рис. 16. Вікно Kibana для перегляду даних підприємства та виявлення НАС

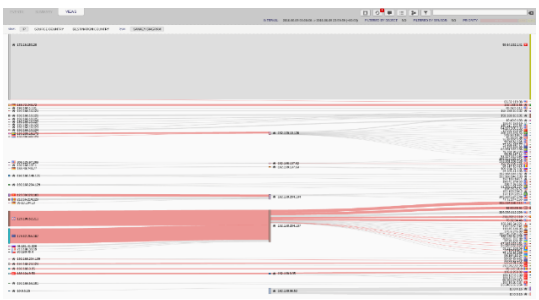

Рис. 17. Фрагмент процесу аналізу та візуалізації мережевих і локальних попереджень за допомогою Squert

Після встановлення відповідного ПЗ користувач отримує комплексне рішення щодо виявлення вторгнень на мережевому і локальному рівнях. В Security Onion поєднуються різні механізми, наприклад, сигнатурний та аномальний підходи, текстовий і графічний інструментарій тощо. Оскільки функціонал системи достатньо великий, то її основним недоліком є значний часовий ресурс необхідний для налаштування ПЗ. Але для пришвидшення роботи користувач може застосувати спрощений функціонал, для якого використовуються не всі програмні засоби [72].

Залежно від попередньо встановленого набору інструментів для виявлення вторгнень, вразливостей та інших дій НАС зазначений засіб працює в активному і пасивному режимах. Завдяки використанню в ПЗ різних детекційних методів та засобів (наприклад, Snort, Suricata, Snorby, Bro тощо), які доповнюють один одного, Security Onion містить систему оповіщення про безпеку та виявлення аномалій і шкідливих програм [80].

Відповідно до засобів аналізу (Kibana, CapME, CyberChef, Squert, ELSA, Sguil), мережевого (Snort, Suricata, Bro, Full Packet Capture) та локального перегляду (Beats, Wazup, Sysmon, Autoruns, Syslog) ПЗ має змогу реагувати на нові загрози (наприклад, шляхом блокування підозрілої IP адреси, з якої надходить велика кількість незнайомого системі трафіка) та заносити їх в особисту базу даних [81].

Відповідно до наявних програмних засобів, дане ПЗ здатне зчитувати різні формати даних та інтегруватися в різні системи [81] (наприклад, CapME може переглядати дані аналізу ПЗ Squert та логів і часових відміток ПЗ Kibana [82]; Squert здатне переглядати HTTP логи, що сформовані ПЗ Bro [83]; ELSA може інтегрувати свої рішення в логи програм Bro, NIDS alerts, OSSEC, syslog, а також інтегруватися в веб-браузери Chromium/Chrome [84] тощо).

Виявлення кібератак на систему відбується за рахунок набору встановлених засобів в ПЗ і засноване на сигнатурних базах даних, статистичних даних та повному контролі змін в системі. Також вбудовані засоби здатні динамічно реагувати на виникнення загроз і їх поведінку [81].

Управління системою може бути централізоване (наприклад, для Snort, Bro, Suricata тощо) і розподілене (наприклад, для OSSEC), а також є можливість адаптування та масштабування відповідно до потреб окремого користувача чи підприємства [78].

За допомогою сукупності програмних засобів спостереження за системою відбувається на системному і мережевому рівнях [27].

Реакція на кібератаку в реальному режимі часу здійснюється тільки у випадку використання в Security Onion відповідного функціоналу, що підтримується необхідним ПЗ із наданого списку [79]. Спеціальні механізми захисту Security Onion не розкриті розробниками, вона підтримується ОС Unix та Linux [27].

Таблиця 1

Зведені дані результатів аналізу СВВ

№	СВВ	Класи кібератак		Адаптивність	Методи виявлення											Управління системою		Масштабованість	Рівень спостереження		Реакція на кібератаку	Захищеність	Підтримка ОС			
		Зловживання	Аномалії		Експертний	Статистичний	Сигнатурний	Графи сценаріїв	Контроль зміни подій	Кластерний	Динамічний	Машинного навчання	Поведінковий	Евристичний	Нечітких множин	Централізоване	Розподілене		Системний	Мережевий			Unix	Linux	Windows	MacOS
1	AAFID	+	+	-	+	-	+	-	-	-	-	-	-	-	-	+	-	+	+	-	-	-	+	+	-	-
2	Snort	+	+	-	-	+	+	-	-	-	-	-	-	-	-	+	-	+	-	+	+	+	+	+	+	-
3	Prelude SIEM	+	+	-	-	-	+	-	-	-	-	-	-	-	-	+	-	+	-	+	+	+	+	+	-	-
4	NetSTAT	+	+	+	-	-	+	-	+	-	-	-	-	-	-	-	+	+	+	+	+	+	+	+	-	-
5	ASAX	+	-	-	+	-	-	-	-	-	-	-	-	-	-	+	-	+	+	-	-	-	+	+	-	-
6	Bro	+	+	-	-	-	+	-	-	-	-	-	-	-	-	+	-	-	-	+	+	-	+	+	-	+
7	OSSEC	+	+	-	-	-	+	-	-	-	-	-	-	-	-	+	+	+	+	-	+	+	+	+	+	+
8	Suricata	+	+	+	-	+	+	-	-	-	-	-	-	-	-	+	-	+	-	+	+	-	+	+	+	+
9	Samhain	+	+	+	-	+	-	-	-	-	+	-	+	-	-	+	+	+	+	-	+	+	+	+	+	+
10	Security Onion	+	+	+	-	+	+	-	+	-	+	-	+	-	-	+	+	+	+	+	+	-	+	+	-	-

Проведений аналіз програмних засобів систем виявлення зловживань та аномалій, за рахунок базових характеристик, таких як клас атак, адаптивність, методи виявлення атак, управління системою, масштабованість, рівень спостереження за системою, реакція на атаку, захищеність та підтримувана ОС, дає можливість для розробників і користувачів обрати відкрите ПЗ для захисту ІС.

Література

[1]. Хакерські атаки на Україну, 2017. [Електронний ресурс]. Режим доступу: https://is.gd/6lkWHY (дата звернення: 17.04.2018).

[2]. Пострадавшие от кибератаки банки и компании: перечень, 2017. [Електронний ресурс]. Режим доступу: https://zn.ua/UKRAINE/postradav-

shiy-ot-kiberataki-banki-i-kompanii-perechen-252717_ .html (дата звернення: 17.04.2018).

[3]. Хакерська атака на Україну: подробиці, 2017. [Електронний ресурс]. Режим доступу: https://www.rbc.ua/ukr/news/hakerskaya-ataka-ukrainu-podrob-nosti-1498566985.html (дата звернення: 17.04.2018).

[4]. А. Мустафаев, "Нейросетевая система обнаружения компьютерных атак на основе анализа сетевого трафика", *Вопросы безопасности*, № 2. С. 1-7, 2016. [Электронный ресурс]. Режим доступа: http://e-notabene.ru/nb/article_18834.html (дата обращения: 18.04.2018).

[5]. А. Корниенко, И. Слюсаренко, "Системы и методы обнаружения вторжений: современное состояние и направления совершенствования", [Электронный ресурс]. Режим доступа: http:// citforum. ru/security/internet/ids_overview/ (дата обращения: 18.04.2018).

[6]. В. Литвинов, "Аналіз систем та методів виявлення несанкціонованих вторгнень у комп'ютерні мережі", *Математичні машини і системи*, № 1, С. 31-40, 2018. [Електронний ресурс]. Режим доступу: URL: https:// cyberleninka.ru/ article/ v/ analiz-sistem-ta-metodiv-vi-yavlennya-nesanktsionovanih-vtorgnen-u-kompyuterni-merezhi (дата звернення: 03.07.2018).

[7]. А. Браницкий, А. Котенко, "Анализ и классификация методов обнаружения сетевых атак", *Тр. СПИИРАН*, № 2 (45), С. 207-244, 2016.

[8]. Краткий анализ решений в сфере СОВ и разработка нейросетевого детектора аномалий в сетях передачи данных, 2018. [Электронный ресурс] Режим доступа: https://habr.com/post/358200/ (дата обращения: 03.07.2018).

[9]. О. Колодчак, "Сучасні методи виявлення аномалій в системах виявлення вторгнень", *Вісник Національного ун-т «Львівська політехніка». Комп'ютерні системи та мережі,* № 745, С. 98-104, 2012.

[10]. Д. Даниленко, О. Смірнов, Є. Мелешко, "Дослідження методів виявлення вторгнень в телекомунікаційні системи та мережі", *Системи озброєння і військова техніка,* Х.: Харк. нац. ун-т Повітряних Сил ім. І. Кожедуба, № 1, С. 92-100, 2012.

[11]. R. Patel, A. Thakkar, A. Ganatra, "A Survey and Comparative Analysis of Data Mining Techniques for Network Intrusion Detection Systems", I*nternational Journal of Soft Computing and Engineering (IJSCE)*, vol. 2, no. 1, pp. 265-260, 2012.

[12]. Al-Sakib Khan Pathan, *The State of the Art in Intrusion Prevention and Detection,* 2014, 516 p. [Electronic reso-urse]. Online: http://docshare03. docshare.tips/files/ 20579/205795770.pdf (viewed on August 4, 2018).

[13]. Г. Бекетова, Б. Ахметов, О. Корченко, В. Лахно, "Розробка моделі інтелектуального розпізнавання аномалій і кібератак з використанням логічних процедур, які базуються на покриттях матриць ознак", *Безпека інформації*, Т. 22, № 3, С. 242-254, 2016.

[14]. К. Носенко, О. Півторак, Т. Ліхоузова, "Огляд систем виявлення атак в мережевому трафіку", *Адаптивні системи автоматичного управління,* К : НТУУ КПІ, № 1 (24), С. 67-75, 2014.

[15]. М. Радченко, "Аналіз системи виявлення вторгнень та комп'ютерних атак", *Междисциплинарные исследования в науке и образовании*, № 2, 2013.

[16]. Amrit Pal Singh, Manik Deep Singh, "Analysis of Host-Based and Network-Based Intrusion Detection System", *I. J. Computer Network and Information Security*, vol. 8, pp. 41-47, 2014.

[17]. В. Мєшков, В. Віролайнен, "Аналіз сучасних систем виявлення та запобігання вторгнень в інформаційно-телекомунікаційних системах", *Проблеми безпеки інформації в інформаційно-комунікаційних системах,* Д.: НТУУ КПІ РТФ, 2015. С. 4. [Електронний ресурс]. Режим доступу: http:// ela. kpi. ua/ bitstream/ 123456789/ 17609/1/meshkov.pdf (дата звернення: 06.07.2018).

[18]. А. Лось, Ю. Даниелян, "Сравнительный анализ систем обнаружения вторжений, представленных на отечественном рынке", *Вестник Московского финасово-юридического универсистета*, № 3. С. 181-187, 2014.

[19]. А. Белова, Д. Бородавкин, "Сравнительный анализ систем обнаружения вторжений", *Актуальные проблемы авиации и космонавтики*, Сибирь: СФУ, Т. 1, № 12, С. 742-744, 2016.

[20]. А. Завада, О. Самчишин, В. Охрімчук, "Аналіз сучасних систем виявлення атак і запобігання вторгненням", *Інформаційні системи*, Житомир: Збірник наукових праць ЖВІ НАУ, Т. 6, № 12, С. 97-106, 2012.

[21]. Обзор систем обнаружения вторжений. Металургический журнал. Отрасли народного хозяйства. Исследования рынка, 2003. [Электронный ресурс] Режим доступа: http://www.metclad.ru/pata-587-list/ (дата обращения: 10.07.2018).

[22]. В. Бабошин, В. Васильев, "Обзор зарубежных и отечественных систем обнаружения компьютерных атак", *Информация и космос.* СПб : Санкт-Петербургская научно-техническая общественная организация «Институт телекоммуникаций», № 2, С. 36-41, 2015.

[23]. С. Гриняев, Системы обнаружения вторжений, № 10, 2001. [Электронный ресурс]. Режим доступа: https://www.bytemag.ru/articles/detail.php?ID=6563 (дата обращения: 10.07.2018).

[24]. Е. Абрамов, И. Половко, "Выбор характеристик систем обнаружения атак для выработки заключения о функциональных возможностях", *Известия Южного федерального университета. Технические науки*. Таганрог : ЮФУ, № 12 (125), С. 88-96, 2011.

[25]. Mohammad Sazzadul Hoque, Md. Abdul Mukit, Md., Abu Naser Bikas, "An implementation of intrusion detection system using genetic algorithm", *International Journal of Network Security & Its Applications (IJNSA)*, Sylhet, Vol. 4, no. 2, pp. 109-120, 2012.

[26]. O. Lawal, "Analysis and Evaluation of Network-Based Intrusion Detectionand Prevention System in an Enterprise Network Using Snort Freeware", *African Journal of Computing & ICT*, Ibadan, Vol. 6, no. 2, pp. 169-184, 2013.

[27]. S. Cooper, 11 Top Intrusion Detection Tools for 2018. [Electronic resource]. Online: https://www.comparitech.com/net-admin/network-intrusion-de-tection-tools/ (viewed on August 12, 2018).

[28]. Т. Зоріна, "Системи виявлення і запобігання атак в комп'ютерних мережах", *Вісник східноукраїнського національного університету імені Володимира Даля*, № 5 (204), С. 48-52, 2013.

[29]. Liu Hua Yeo, Understanding modern intrusion detection systems: a survey, 2017. [Electronic resource]. Online: https:// arxiv. org/ ftp/ arxiv/ papers/ 1708/ 1708.07174.pdf (viewed on August 12, 2018).

[30]. Д. Гамаюнов, Р. Смелянский, "Современные некоммерческие средства обнаружения атак", *Программные системы и инструменты. Тематический сборник.* М. : Ф-т ВМиК МГУ, С. 20, 2002.

[31]. А. Корниенко, И. Слюсаренко, "Системы и методы обнаружения вторжений: современное состояние и направления совершенствования", 2009. [Электронный ресурс]. Режим доступа: http://citforum. ru/ security/ internet/ ids_overview/ (дата обращения: 15.07.2018).

[32]. Е. Явтуховский, "Анализ систем обнаружения вторжений на основе интеллектуальных технологий", *Технические науки: теория и практика: материалы III Междунар. науч. конф.*, С. 27-30, 2016. [Электронный ресурс]. Режим доступа: https:// moluch. ru/ conf/ tech/archive/165/10049/ (дата обращения: 17.07.2018).

[33]. A. Kuznetsov, "The statistical analysis of a network traffic for the intrusion detection and prevention systems", *Telecommunications and Radio Engineering*, Kharkiv, vol. 74, no. 1, 2015.

[34]. Marjan Kuchaki Rafsanjani, Zahra Asghari Varzaneh, "Intrusion Detection By Data Mining Algorithms: A Review", *Journal of New Results in Science*, Tokat : Gaziosmanpasa University, no. 2. pp. 76-91, 2013.

[35]. О. Кузнецов, О. Смірнов, Д. Даниленко, "Дисперсійний аналіз мережевого трафіку для виявлення та запобінання вторгнень в телекомунікаційних системах і мережах", *Системи обробки інформації*, Х. : Харк. нац. ун-т Повітряних Сил ім. І. Кожедуба, Вип. 2, С. 124-133, 2014.

[36]. Neyole Misiko Jacob, Muchelule Yusuf Wanjala, "A Review of Intrusion Detection Systems", *Global Journal of Computer Science and Information Technology Research. Framingham : Global Journals Inc.*, Vol. 5, no. 4, pp. 1-5, 2017.

[37]. А. Большев, В. Яновский, "Подход к обнаружению аномального трафика в компьтерных сетях с использованием методов кластерного анализа", *Известия Государственного Электротехнического Университета, серия Информатика, управления и компьютерные технологии,* СПб. : Изд-во СПбЭТУ, Вып. 3. С. 38-45, 2006.

[38]. А. Корченко, С. Ахметова, "Классификация систем обнаружения вторжений", *Інформаційна безпека*. № 1 (13); № 2 (14). С. 168-175, 2014..

[39]. В. Мєшков, В. Віролайнен, "Аналіз сучасних систем виявлення та запобігання вторгнень в інформаційно-телекомунікаційних системах", *Проблеми безпеки інформації в інформаційно-комунікаційних системах*. К. : НТУУ КПІ РТФ, №. 1. С. 1-4, 2015.

[40]. М. Грайворонський, О. Новіков, *Безпека інформаційно-комунікаційних систем : навч. посіб,* К. : Видавнича група BHV, 2009, 608 с.

[41]. А. Корченко, *Построение систем защиты информации на нечетких множествах. Теория и практические решения,* К. : МК-Пресс, 2006, 320 с.

[42]. О. Матов, В. Василенко, "Модель загроз у розподілених мережах", *Реєстрація, зберігання та обробка даних,* К. : НАУ, Т. 10, № 1. С. 91-102, 2008.

[43]. Security Research Laboratory and Education Center, 1999. [Electronic resource]. Online: https:// www.linuxjournal.com/article/3175 (viewed on August 20, 2018).

[44]. E. Spafford, D. Zamboni, CERIAS – Autonomous Agents for Intrusion Detection, 2000. [Electronic resource]. Online: http://www.cerias.purdue.edu/site/about/history/coast/projects/aafid.php (viewed on August 20, 2018).

[45]. J. Balasubramaniyan, An architecture for intrusion detection using autonomous agents, 2002. [Electronic resource]. Online: https://ieeexplore.ieee.org/abstract/document/738563 (viewed on August 20, 2018).

[46]. J. Balasubramaniyan, An Architecture for Intrusion Detection using Autonomous Agents, 1998. [Electronic resource]. Online: https://pdfs.semanticschoar.org/bb4b/a3a4e8b850011844c00aa0fa964bf4664b23.pdf (viewed on August 20, 2018).

[47]. SNORT. Snort team. San Jose: Cisco Systems Inc, 2018. [Electronic resource]. Online: https:// www.snort.org/ (viewed on August 23, 2018).

[48]. IDS / IPS. Netgate Documentation: [website]. Washington : Rubicon Communications LLC, 2017. [Electronic resource]. Online: https://www.netgate.com/docs/pfsense/ids-ips/ (viewed on August 23, 2018).

[49]. Джей Бил, *Snort 2.1. Обнаружение вторжений : книга*, 2006, 656 с.

[50]. Snort. Spy-Soft.net: Информационная безопасность на практике, 2016. [Электронный ресурс]. Режим доступа: http://www.spy-soft.net/snort/ (дата обращения: 24.07.2018).

[51]. Snort. Snort team. Snort Blog : the Official Blog of the World Leading Open-Source IDS/IPS Snort : [website]. San Jose : Cisco Systems Inc, 2017. [Electronic resource]. Online: https:// blog. snort. org/ 2017/ 10/ snort-29110-has-been-released.html (viewed on August 24, 2018).

[52]. Prelude SIEM. Prelude SIEM. CS Communication & Systemes, 2018. [Electronic resource]. Online: https://www.prelude-siem.org/ (viewed on August 26, 2018).

[53]. SIEM на практике: дружим Prelude + Cisco IPS и выявляем эксплуатацию HeartBleed через корреляцию, 2014. [Электронный ресурс]. Режим доступа: https://habr.com/post/220449/ (дата обращения: 26.07.2018).

[54]. S. Eckmann, G. Vigna, R. Remmerer, "STATL: An Attack Language for State-based Intrusion Detection", *Journal of Computer Security, Santa Barbara*, pp. 1-29, 2000.

[55]. G. Vigna, R. Kemmerer, "NetSTAT: A Network-based Intrusion Detection Approach", *Proceedings 14th Annual Computer Security Applications Conference. Phoenix : IEEE*, pp. 1-10, 1998.

[56]. Koral Ilgun, "USTAT: A real-time intrusion detection system for UNIX", *Proceedings 1993 IEEE Computer Society Symposium on Research in Security and Privacy. Oakland : IEEE*, pp. 16-28, 1993.

[57]. Naji Habra, Baudouin Le Charlier, Abdelaziz Mounhji, Isabelle Mathieu, "ASAX: Software architecture and rule-based language for universal audit trail analysis", *Proceedings of ESORICS`92 European Symposium on Research in Computer Security.* Toulouse, Vol. 648. pp. 435-450, 1992.

[58]. A. Mounji, "Preliminary Report on Distributed ASAX", *Research Report, Computer Science Institute.* Namur : University of Namur, 1994.

[59]. N. Habra, B. Le Charlier, A. Mounji, "Advanced Security Audit Trail Analysis on Unix. Implementation Design of the NADF Evaluator", *Technical report*. Namur : University of Namur, 1993.

[60]. N. Habra, B. Le Charlier, A. Mounji, "Advanced Security Audit Trail Analysis on Unix (ASAX also called SAT-X). Implemention design of the NADF Evaluator", 1994. [Electronic resource]. Online: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.51.7971&rep=rep1&type=pdf (viewed on September 3, 2018).

[61]. Р. Ярыженко, "Большой Брат: обзор системы обнаружения вторжений Bro" [Электронный ресурс]. Режим доступа: https://xakep.ru/2015/06/28/big-bro-197/ (дата обращения: 05.09.2018).

[62]. Vern Paxson, "Bro: A system for detecting network intruders in real-time", *Proceedings of the 7th USENIX Security Symposium*. San Antonio : USENIX, 1998, 22 p.

[63]. Vern Paxson, "Bro: A system for detecting network intruders in real-time", *Computer Networks. Amsterdam : Elsevier*, no. 31 (23-24), pp. 2435-2463, 1999.

[64]. Critique of Article Bro: A system for Detecting Network Intruders in Real-Time [Electronic resourse]. Online: https://www.24houranswers.com/college-homework-library/Computer-Science/Network-Management-and-Data-Communication/25914 (viewed on September 6, 2018).

[65]. Vern Paxson, *Zeek.* [Electronic resource]. Online: https:// www. bro. org/ download/ index.html (viewed on September 6, 2018).

[66]. James Nelson, "Installing the Splunk for OSSEC App", 2012. [Electronic resource]. Online: http://grepthelinuxblog.blogspot.com/2012/03/installing-splunk-for-ossec-app.html (viewed on September 8, 2018).

[67]. Home-OSSEC, 2018. [Electronic resource]. Online: https://www.ossec.net/ (viewed on September 8, 2018).

[68]. Downloads-OSSEC, 2018. [Electronic resource]. Online: https://www.ossec.net/downloads.html (viewed on September 8, 2018).

[69]. OSSEC and attacking through the firewall, 2016. [Electronic resource]. Online: https://www.cs.hioa.no/teaching/materials/MS004A/html/L65.en.pdf (viewed on September 8, 2018).

[70]. O. Ahmet, "OSSEC-HIDS. Capabilities, Architecture and plans", *Presentation at the 5th Linux and Free Software Festival*. Ankara, 2006.

[71]. Suricata | Open Source IDS/IPs/NSM engine, 2018. [Electronic resource]. Online: https://suricata-ids.org/ (viewed on October 10, 2018).

[72]. Top 10 Intrusion Detection Tools: Your Best Free Options for 2019, 2018. [Electronic resource]. Online: https://www.addictivetips.com/net-admin/ intrusion-detection-tools/ (viewed on October 11, 2018).

[73]. Suricata как IPS. [Электронный ресурс]. Режим доступа: https://habr.com/post/192884/ (дата обращения: 11.10.2018).

[74]. Мартин Пранкевич, День сурка. Осваиваем сетевую IDS/IPS Suricata [Электронный ресурс]. Режим доступа: https://xakep.ru/2015/06/28/suricata-ids-ips-197/ (дата обращения: 11.10.2018).

[75]. Rainer Wichmann, The SAMHAIN file integrity / host-based intrusion detection system. [Electronic resource]. Online: https://www.la-samhna.de/samhain/index.html (viewed on October 14, 2018).

[76]. Examining Tripwire And Samhain IDS Files Information Technology Essay. [Electronic resource]. Online: https://www.ukessays.com/essays/information-technology/examining-tripwire-and-samhain-ids-files-information-technology-essay.php (viewed on October 14, 2018).

[77]. Rainer Wichmann, The SAMHAIN file integrity / host-based intrusion detection system. [Electronic resource]. Online: https://la-samhna.de/samhain/s_faq.html (viewed on October 14, 2018).

[78]. Phil Plantamura, Security Onion [Electronic resourse]. Online: https://securityonion.net/ (viewed on October 16, 2018).

[79]. Phil Plantamura, Security Onion Solutions. [Electronic resource]. Online: https://security-onionsolutions.com/ (viewed on October 16, 2018).

[80]. Security Onion – Intrusion Detection and Network Security Monitoring. [Electronic resource]. Online: https://honim.typepad.com/biasc/2017/12/security-onion-.html (viewed on October 16, 2018).

[81]. IntroductionToSecurityOnion [Electronic resource]. Online: https://github.com/Security-Onion-Solutions/security-onion/wiki/IntroductionToSecurity Onion (viewed on Octiber 16, 2018).

[82]. CapMe [Electronic resource]. Online: https://github.com/Security-Onion-Solutions/security-onion/wiki/CapMe (viewed on Octiber 16, 2018).

[83]. Squert [Electronic resource]. Online: https://github.com/Security-Onion-Solutions/security-onion/wiki/Squert (viewed on Octiber 16, 2018).

[84]. ELSA [Electronic resource]. Online: https://github.com/Security-Onion-Solutions/security-onion/wiki/ELSA (viewed on Octiber 16, 2018).

УДК 004.056.53(045)

Терейковский И., Корченко А., Паращук Т., Педченко Е. Анализ открытых систем обнаружения вторжений

Аннотация. *Постоянное развитие информационных систем влияет на все сферы деятельности общества. Одним из актуальных направлений, которое активно развивается в сфере информационной безопасности является выявление кибератак и предотвращение вторжений. Массовые кибератаки инициируют создание специальных технических решений, средств и систем противодействия. Для обнаружения сетевых вторжений используются современные методы, модели, средства, программное обеспечение и комплексные технические решения для систем обнаружения и предотвращения вторжений, которые могут оставаться эффективными при появлении новых или модифицированных видов киберугроз. На практике при появлении новых угроз и аномалий, указанные средства не всегда остаются эффективными. Поэтому системы обнаружения вторжений должны постоянно исследоваться и совершенствоваться. Среди таких систем есть специализированные программные средства, направленные на выявление подозрительной активности или вмешательства в информационную систему и принятия адекватных мер по предотвращению кибератак. Эти системы и средства, как правило, достаточно дорогие, имеют закрытый код и требуют периодической поддержки разработчиков по их усовершенствованию и соответствующей настройки к условиям конкретных организаций. Учитывая результаты известных исследований в работе проведен обобщенный анализ программных средств систем обнаружения вторжений с определенным базовым множеством характеристик («Класс кибератак», «Адаптивность», «Методы выявления», «Управление системой», «Масштабируемость», «Уровень наблюдения», «Реакция на кибератаки», «Защищенность» и «Поддержка операционной системы»). Это даст определенные возможности для разработчиков и пользователей выбрать соответствующее современное программное обеспечение для защиты информационных систем.*

Ключевые слова: *атаки, кибератаки, аномалии, злоупотребления, системы обнаружения вторжений, системы обнаружения кибератак, системы обнаружения аномалий, выявление аномалий в информационных системах.*

Tereykovsky I., Korchenko A., Parashchuk T., Pedchenko Y., Open intrusion detection systems analysis

Abstract. *Ongoing advances in information technology affect all areas of society. One of the most promising areas of rapid growth within the field of information security is cyberattack detection and intrusion prevention. Massive cyberattacks initiate the development of specific technical solutions, tools and cyber countermeasures systems. To identify network intrusions, intrusion detection and prevention systems use modern methods, models, tools and integrated technical solutions that can remain effective when new or modified types of cyberthreats occur. In practice, however, with the emergence of new threats and anomalies, these tools do not always remain effective. Thus, intrusion detection systems must be continuously researched and improved. Such systems include specialized software that is designed to detect suspicious activity or information system intrusions and take sufficient measures to prevent cyberattacks. These systems and tools tend to be rather expensive, closed source, and require periodic support from their developers for improvement and appropriate adaptation to certain organizations' environments. Taking into account the results of well-known research, the paper presents a generalized analysis of intrusion detection systems software using a defined basic set of characteristics ("Cyberattack Category", "Adaptivity", "Detection Methods", "System Management", "Scalability", "Observation Level", "Cyberattack Response", "Security" and "Operating System Support"). This will provide the developers and users with certain options when selecting the appropriate modern information systems protection software.*

Keywords: *attacks, cyberattacks, anomalies, exploits, intrusion detection systems, cyberattack detection systems, anomaly detection systems, information systems anomaly detection.*

Отримано 26 *листопада 2018 року*, затверджено редколегією 10 *грудня 2018 року*