DOI 10.21672/2074-1707.2020.52.4.085-098
УДК 004.056

ИССЛЕДОВАНИЕ ПРИМЕНЕНИЯ ТЕХНОЛОГИИ DECEPTION ДЛЯ ПРЕДОТВРАЩЕНИЯ УГРОЗ КИБЕРБЕЗОПАСНОСТИ

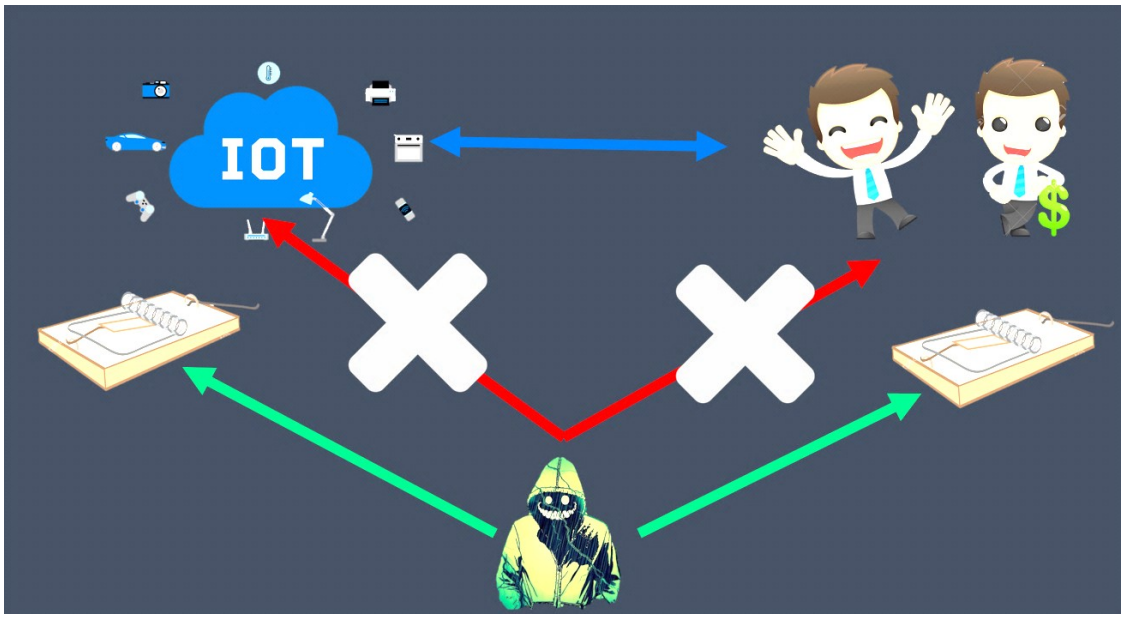
Статья поступила в редакцию 15.10.2020, в окончательном варианте – 25.10.2020.

Путято Михаил Михайлович, Кубанский государственный технологический университет, 350072, Российская Федерация, г. Краснодар, ул. Московская, 2,
кандидат технических наук, доцент, e-mail: putyato.m@gmail.com
Макарян Александр Самвелович, Кубанский государственный технологический университет, 350072, Российская Федерация, г. Краснодар, ул. Московская, 2,
кандидат технических наук, доцент, e-mail: msanya@yandex.ru
Чич Шамиль Муратович, Кубанский государственный технологический университет, 350072, Российская Федерация, г. Краснодар, ул. Московская, 2,
студент, e-mail: shama_chich@icloud.com
Маркова Валентина Константиновна, Кубанский государственный технологический университет, 350072, Российская Федерация, г. Краснодар, ул. Московская, 2,
студентка, e-mail: markokovt@yandex.ru

Устройства интернета вещей (IoT) приобретают все большую популярность за последнее время. Под IoT подразумевают умные холодильники, умные замки, видеоняни и иные бытовые устройства, имеющие выход в интернет. Однако рост популярности технологии IoT все больше привлекает внимание злоумышленников, заинтересованных как в раскрытии конфиденциальных данных конечных пользователей, так и в нецелевом использовании вычислительных ресурсов атакуемых устройств. К сожалению, атаки злоумышленников зачастую завершаются успешной компрометацией устройств с вытекающими последствиями. Причины высокого уровня компрометации IoT-устройств вызваны как ошибками при проектировании, реализации, так и относительно простой возможностью эксплуатации с использованием различных средств аудита информационной безопасности. Для выявления дефектов разработки и реализации устройств необходимо иметь какое-то представление о них, то есть своевременно выявлять и устранять. Этого можно добиться различными способами. Одним из таких способов является создание специальных ловушек, собирающих информацию об активности злоумышленника, называемых honeypot. Суть технологии honeypot заключается в эмуляции или имплементации функционала существующих устройств, сервисов, протоколов с накоплением данных о вредоносной активности злоумышленника. Полученную информацию можно использовать как для улучшения защиты реальных устройств, сервисов, протоколов, так и для разработки мер противодействия злоумышленникам. В статье проводится сравнительный анализ существующих наиболее популярных honeypot-систем с целью выявления наиболее лучшей системы. В ходе анализа удалось выделить как слабые, так и сильные стороны этих систем. Далее проводится попытка адаптации этих же систем для функционирования на уровне устройств интернета вещей.

Ключевые слова: кибербезопасность, deception technology, honeypot, форензика, вредоносная программа, хакерская активность, информационная безопасность, кибер-ловушка, honeynet, защита информации

Графическая аннотация (Graphical annotation)



RESEARCH ON THE USE OF DECEPTION TECHNOLOGY TO PREVENT CYBERSECURITY THREATS

The article was received by the editorial board on 15.10.2020, in the final version – 25.10.2020.

Putyato Mikhail M., Kuban State Technological University, 2 Moskovskaya St., Krasnodar, 350072, Russian Federation,
Cand. Sci. (Engineering), Associate Professor, e-mail: putyato.m@gmail.com
Makaryan Aleksandr S., Kuban State Technological University, 2 Moskovskaya St., Krasnodar, 350072, Russian Federation,
Cand. Sci. (Engineering), Associate Professor, e-mail: msanya@yandex.ru
Chich Shamil M., Kuban State Technological University, 2 Moskovskaya St., Krasnodar, 350072, Russian Federation,
student, e-mail: shama_chich@icloud.com
Markova Valentina K., Kuban State Technological University, 2 Moskovskaya St., Krasnodar, 350072, Russian Federation,
student, e-mail: markokovt@yandex.ru

Internet of things (IoT) devices have become increasingly popular in recent years. IoT refers to smart refrigerators, smart locks, video nannies, and other household devices that have access to the Internet. However, the growing popularity of IoT technology is increasingly attracting the attention of hackers who are interested both in disclosing confidential end-user data and in misuse of the computing resources of the attacked devices. Unfortunately, malicious attacks often result in successful compromise of devices, with the ensuing consequences. The reasons for the high level of compromise of IoT devices are caused both by errors in the design, implementation, and relatively simple operation with the use of various information security audit tools. To identify defects in the development and implementation of devices, you need to have some idea about them, that is, to identify and eliminate them in a timely manner. This can be achieved in various ways. One of these methods is to create special traps that collect information about the activity of an attacker, called honeypot. The essence of the honeypot technology is to emulate or implement the functionality of existing devices, services, and protocols, with the accumulation of data about malicious activity of an attacker. The information obtained can be used to improve the protection of real devices, services, and protocols, as well as to develop measures to counter hackers. The article provides a comparative analysis of the existing most popular honeypot systems in order to identify the best system. The analysis identified both the weaknesses and strengths of these systems. Next, an attempt is made to adapt these same systems to function at the level of Internet of things devices.

Keywords: cybersecurity, deception technology, honeypot, malicious ware, forensics malware, hacker activity, information security, cyber trap, honeynet, information security

Введение. Одна из основных задач, которую приходится решать специалистам по информационной безопасности, – это сбор сведений, позволяющих обнаружить злоумышленников, понять их цели и мотивы, обнаружить и пресечь их действия. Раньше суть киберугрозы пытались выяснить, исключительно анализируя вредоносное программное обеспечение, использованное для проникновения: после того как произошел инцидент, единственные данные, которыми располагали специалисты, – это информация, остававшаяся во взломанной системе. К тому же, в последнее время наблюдается тенденция к так называемым бестелесным вредоносным программам. Все вредоносные действия такого класса программ производятся в ОЗУ, без записи исполняемого файла на диск, что делает затруднительным и даже невозможным анализ методов работ злоумышленников. Однако даже в случае успешного получения исполняемого файла программы, ситуация вряд ли поменяется, так как данные были скомпрометированы и/или безвозвратно потеряны.

Deception-технология – это использование техник активного обмана атакующих с применением специализированных ловушек, приманок и других методов дезинформации. Применение техник обмана внутри корпоративного периметра предоставляет предприятиям возможность раннего обнаружения наиболее опасных направленных атак, которые не были отслежены превентивными механизмами, такими как межсетевые экраны, системы предотвращения вторжений и антивирусные решения. Несмотря на то, что сам термин Deception еще достаточно новый, в основе технологии лежит известная концепция Honeypot (ловушка), но с принципиально другим уровнем реализации, возможностями масштабирования и автоматизации [9].

Deception-платформы – это централизованно управляемые системы для организаций, предназначенные для создания, распространения и управления всей обманчивой средой и связанными с ней архитектурными элементами, такими как подставные рабочие станции, серверы, устройства, приложения, службы, протоколы, элементы данных или пользователи, которые часто виртуализируются и по существу неотличимы от реальных активов и используются в качестве приманки для привлечения и обнаружения злоумышленника. Принцип работы для класса описанных платформ показан на рисунке 1.

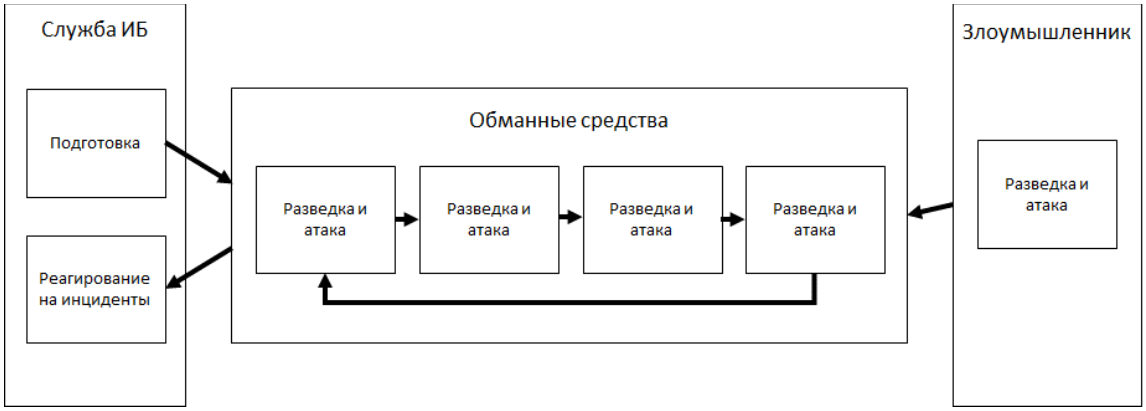


Рисунок 1 – Принцип работы deception-систем

Обзор систем класса distributed deception platform. На сегодняшний день каждая из задач обеспечения кибербезопасности решается огромным числом различных систем, в числе которых: NGFW (Next Generation Firewall), SIEM [11] (Security Incident and Event Management), EPP (Endpoint Protection Platform), EDR (Endpoint Detection and Response) и т.д. [10].

Каждая из этих систем нацелена на прямую защиту существующих ИТ активов организации с помощью непосредственного встраивания между целевой системой и атакующим. Главное предположение, исходя из которого строится защита, является безусловная уверенность в том, что атака проводится на реальный ИТ актив. Одной из таких технологий защиты является Distributed Deception Platform (DDP) или распределённые платформы для имитации инфраструктуры [10].

В таблице 1 приведено сравнение современных DDP, основанное на таблице сравнения Deception Techniques [13].

Таблица 1 – Сравнительный анализ существующих deception-платформ

	TrapX Deceptiongrid Platform	CYMMETRIA'S MazeRunner	CYBERTRAP	ATTIVO NETWORKS ThreatDefend Platform	GuardiCore Centra Security Platform	ILLUSIVE Platform	Acalvio shadowplex	FIDELIS Elevate	Illusion BLACK
Deception токены (подделные ОС)	Windows Linux	Windows	Windows	Windows Linux Mac		Windows	Windows	Windows	Windows
Интеграция веб-приложений			+						+
Обнаружение командных центров, MITM и бот-сетей	Обнаружение командных центров + обнаружение атак MITM + ботнет-детектор	Обнаружение атак MITM		Обнаружение командных центров + Обнаружение атак MITM + ботнет-детектор				Обнаружение командных центров + ботнет-детектор	
Этапы обнаружения атак	Разведка Боковое движение Эксфильтрация	Разведка Боковое движение Эксфильтрация	Разведка Боковое движение Эксфильтрация	Разведка Боковое движение Эксфильтрация		Разведка Боковое движение	Разведка Боковое движение Эксфильтрация	Разведка Боковое движение Эксфильтрация	Разведка Боковое движение Эксфильтрация

Продолжение таблицы 1

Приманки для специфической отрасли	+			+		+			
NAC интеграция	+			+		+	+		
Тип ловушек	Ловушки на базе полноценной ОС + наличие эмулированных сенсоров	Ловушки на базе полноценной ОС + наличие эмулированных сенсоров	Ловушки на базе полноценной ОС	Ловушки на базе полноценной ОС + наличие эмулированных сенсоров	Ловушки на базе полноценной ОС	Ловушки на базе полноценной ОС	Ловушки на базе полноценной ОС + наличие эмулированных сенсоров	Наличие эмулированных сенсоров	Ловушки на базе полноценной ОС + наличие эмулированных сенсоров
Корреляция результатов	SIEM интеграция + встроенные средства корреляции	Встроенные средства корреляции		SIEM интеграция + встроенные средства корреляции	SIEM интеграция + встроенные средства корреляции	SIEM интеграция + встроенные средства корреляции	SIEM интеграция + встроенные средства корреляции	SIEM интеграция + встроенные средства корреляции	SIEM интеграция
Интеграция Endpoint	+			+		+	+	+	+
EDR	+			+			+		
Оркестровка	+			+			+		
Active Directory	+			+		+	+	+	+
База данных	+		+	+			+		+
Общий сетевой ресурс	+			+			+		+
Облачные среды	AWS Azure OpenStack	+	Доступен SaaS	AWS Azure OpenStack GCP	AWS Azure OpenStack Доступен SaaS		AWS Azure OpenStack		
Адаптация под нужды пользователя	Конструктор пользовательских ловушек + использование пользовательских образов			Конструктор пользовательских ловушек + использование пользовательских образов		Использование пользовательских образов	Использование пользовательских образов		
API	Наличие открытого API для интеграции + REST API			REST API			Наличие открытого API для интеграции + REST API		
Возможности анализа вредоносного ПО	Интеграция Sandbox + Автоматический анализ кода			Интеграция Sandbox			Интеграция Sandbox		
Интеграция с промышленными системами	POS ATM SCADA IoT	IoT		POS SCADA IoT			SCADA IoT	SCADA	IoT

Сравнивая между собой распределенные deception-платформы, можно заключить, что наибольшими функциональными возможностями обладает DDP-система TrapX Deceptiongrid Platform.

Обзор систем Honeypot. Технология honeypot представляет из себя кибер-ловушку, включающую в себя одну или несколько уязвимостей программного обеспечения, то есть является приманкой для злоумышленников. Теоретически идея скрытых ловушек была придумана ещё давно, но в настоящее время существует не так много практических решений. Основное преимущество данной технологии в том, что её можно настроить практически под любую систему.

Honeypot может выполнять различный спектр задач:

1) определять начало атаки;
2) предотвращать атаки;
3) оказывать содействие в расследовании инцидентов информационной безопасности;
4) собирать информацию о действиях злоумышленников;

Существует 2 вида ловушек:

1) исследовательские. Помогают собирать данные о злоумышленниках;
2) промышленные. Предназначены для обнаружения, предотвращения атак и помощи в расследовании инцидентов информационной безопасности.

В технологии Honeypot есть критерий интерактивности, которая определяет возможности злоумышленника при успешной компрометации системы. Чем больше интерактивность, тем больше возможностей у злоумышленника. Как правило, в исследовательских ловушках интерактивность гораздо выше, чем в промышленных.

Преимущества технологии honeypot:

1. Honeypot не требовательны к ресурсам сети, так как отвечают только на запросы, направленные непосредственно ей.
2. Honeypot в случае компрометации способны предоставить отпечаток всех действий злоумышленника.
3. Данной технологии характерны сравнительно низкие показатели вероятностей отказов и ложных срабатываний.

Однако, как и любая технология, Honeypot имеет и свои недостатки:

1. Настройка параметров Honeypot довольно сложная техническая задача, которую приходится решать вручную.
2. Для данной технологии свойственна пассивность, так как злоумышленник должен сам выбрать цель для атаки.
3. Использование Honeypot предполагает выделение некоторого количества ресурсов сети.

Общая архитектура сети приманок honeynet приведена на рисунке 2.

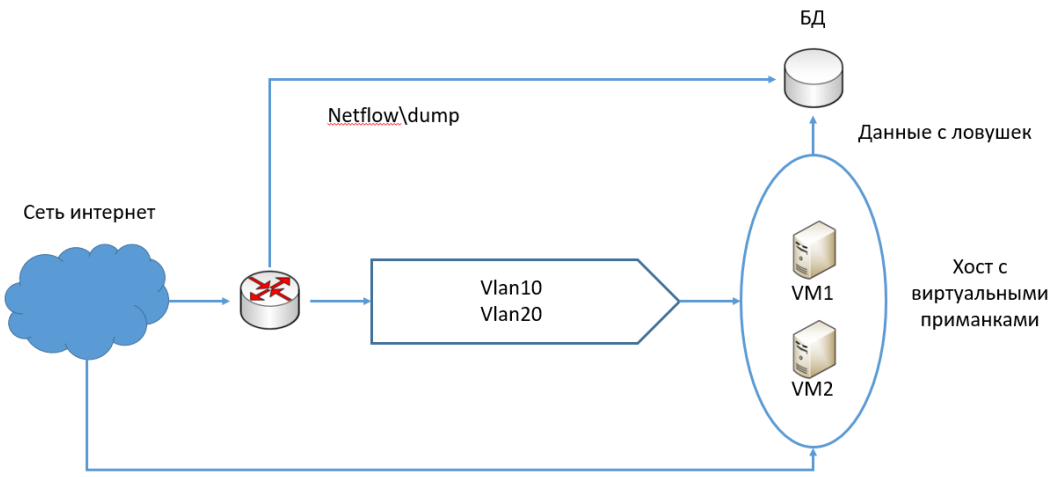


Рисунок 2 – Общая архитектура Honeynet

На данном рисунке вместо множества произвольного числа виртуальных машин VM1, VM2 могут быть реальные физические устройства, порты, сервисы. Весь сетевой трафик, приходящий на сеть с honeypot, записывается в базу данных, также в базу данных записываются данные с приманок. Для обеспечения большей безопасности ловушки располагаются в отдельном сегменте сети.

Исторически первой архитектурой honeynet была сеть Gen I. Данную архитектуру возможно использовать только в исследовательских целях для изучения методов работы злоумышленников. Схема данной сети приведена на рисунке 3.

Хронологическим продолжением сети Gen I стала сеть Gen II. Для Gen II характерно разделение сети на 2 сегмента: сегмент внутренней сети с рабочими сервисами, АРМ, серверами и сегмент сети приманок. Стоит выделить сенсор honeypot, который регистрирует вредоносную активность и совместно с маршрутизатором перенаправляет вредоносный трафик в сегмент сети приманок. Архитектура Gen II представлена на рисунке 4.

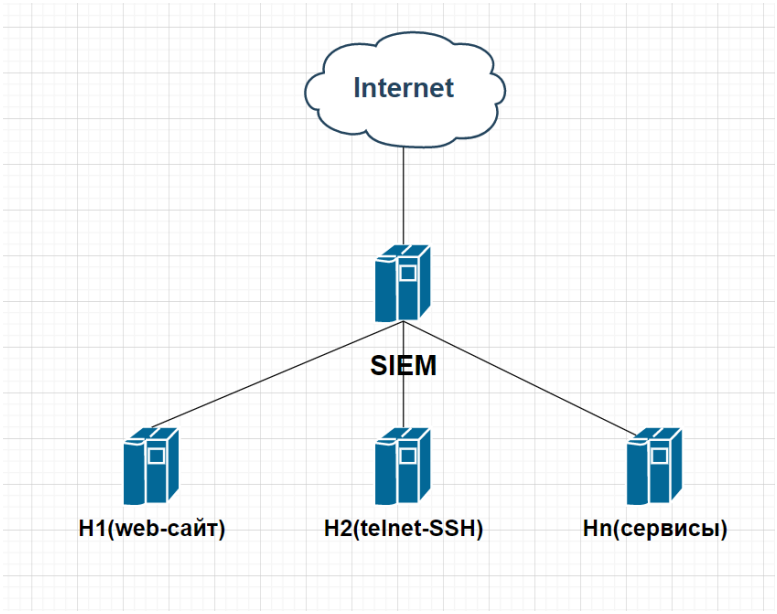


Рисунок 3 – Архитектура сети Gen I

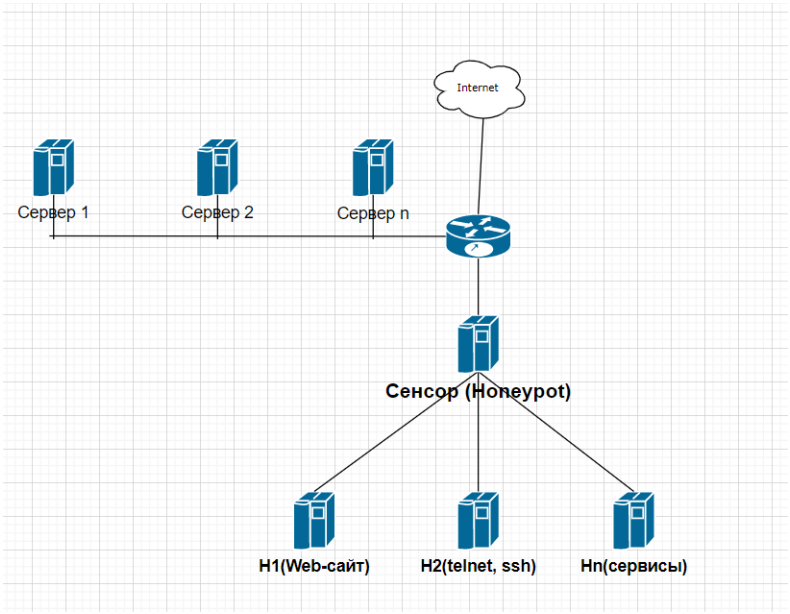


Рисунок 4 – Схема Gen II

Благодаря сенсору Honeypot в архитектуре Gen II становится возможным перенаправление вредоносного трафика в сегмент сети с Honeypot-системами. В отличие от Gen I, архитектуру данной системы можно использовать не только в научно-исследовательских целях, но и интегрировать в уже функционирующие промышленные системы. То есть данный Honeynet приобретает функциональные возможности deception-систем. Однако данная архитектура не получила широкого распространения из-за сложностей, связанных с обнаружением вредоносного трафика.

Исследование. В настоящий момент существуют ловушки как с открытым исходным кодом, так и закрытые. Произведен сравнительный анализ наиболее популярных ловушек с открытым исходным кодом по следующим критериям [1]:

1. *Процесс установки и настройки* – характеризует время и усилия настройки системы. То есть чем сложнее Honeypot и чем обширнее задачи, которые он выполняет, тем выше данный параметр. Простой процесс установки не включает в себя требований по конфигурированию или заданию дополнительных параметров работы средства Honeypot, средний – требует установки или выбора ограниченного числа параметров (например, выбор варианта имитируемого сервиса) для корректной работы средства, сложный – необходима достаточно детальная настройка (обычно сопряженная с установкой), задание большого числа дополнительных параметров функционирования средства Honeypot.

2. *Процесс использования и поддержки* – характеризует время и усилия при использовании и поддержке Honeypot после процесса установки и настройки. Очевидным образом следует, что чем выше функциональность Honeypot, тем сложнее его использование, и тем больше времени и усилий требует его поддержка. Простой процесс использования и поддержки подразумевает минимальное количество времени, необходимое для поддержки средства, а также простоту в использовании (например, для корректного использования необходимо лишь запустить программу). Средний уровень добавляет в процесс использования дополнительные действия, направленные на настройку Honeypot в процессе функционирования, необходимость участия администратора для корректировки поведения и т.п. Сложный уровень включает в себя необходимые действия в использовании и последующей поддержке средства (например, обновление программного обеспечения, восстановление окружения для дальнейшего исследования после того, как было произведено взаимодействие).

3. *Уровень протоколирования* – характеризует степень детализации, с которой будет произведено протоколирование. Чем выше уровень протоколирования, тем большую детализацию имеют записи протокола программы. Низкий уровень протоколирования определяется небольшим уровнем подробности собранных данных. Как правило, данным уровнем обладают средства Honeypot слабого взаимодействия, когда протоколируются только IP-адрес источника взаимодействия и данные, исходящие от злоумышленника. Средний уровень протоколирования может включать в себя протокол обеих сторон взаимодействия, а также дополнительные данные (например, конкретное время прихода данных, идентификаторы взаимодействий и т.п.). Высоким уровнем протоколирования обладают Honeypot сильного взаимодействия, когда средство берет на себя обязанность протоколировать все события, происходящие в системе при взаимодействии.

4. *Уровень имитации* – характеризует степень имитации сервиса. Простой уровень имитации подразумевает практически полное отсутствие поддержки функциональности имитируемого сервиса (например, возможность выводить только приветственное сообщение при соединении). Средний уровень подразумевает достаточно подробную имитацию сервиса с учетом особенностей его работы. Высокий уровень имитации предполагает полную реализацию всех функциональных возможностей сервиса (по сути, происходит приближение к эмуляции сервиса). При использовании реальных операционных систем уровень имитации – высокий. В данную характеристику также включается уровень имитации поведения при реакции на попытку атаки.

5. *Уровень риска* – характеризует степень риска при использовании Honeypot. Чем больше функциональных возможностей предоставляет Honeypot, тем выше вероятность, что Honeypot может быть использован для атаки других систем или сервисов. Низкий уровень риска включает в себя возможную атаку только против имитируемого сервиса. Средним уровнем риска обладают средства Honeypot, имитирующие несколько сервисов одновременно. Высокий уровень риска – недостаток Honeypot сильного взаимодействия.

Сравнивались следующие Honeypot-системы:

1. *Cowrie.* Cowrie – это SSH и Telnet honeypot со средним и высоким уровнем взаимодействия, предназначенные для регистрации атак методом перебора и взаимодействия злоумышленника с оболочкой командной строки. В режиме среднего взаимодействия (shell) он эмулирует систему UNIX на Python, в режиме высокого взаимодействия (proxy) он функционирует как SSH и Telnet прокси для наблюдения за поведением злоумышленника в другой системе.

Возможности:

- поддельная файловая система с возможностью добавления и удаления файлов;
- возможность добавления поддельных файлов, чтобы дать злоумышленнику возможность взаимодействия с ним;
- использование как прокси-сервера для повышения интерактивности. Ловушка выступает в роли монитора и собирает сведения об активности злоумышленника.

2. *Kippo.* Kippo – это SSH-ловушка со средней интерактивностью, предназначенная для регистрации атак методом перебора и взаимодействия злоумышленника с оболочкой.

3. *Glastopf.* Данный Honeypot является веб-приложением. Реализован на языке Python версии 2.7, имеет открытый исходный код, доступный по ссылке https://github.com/mushorg/glastopf. Приложение, которое имитирует данная ловушка, может быть дописано или вовсе переделано в зависимости от нужд пользователей.

4. *Google Hack Honeypot.* Старый проект от корпорации Google. В настоящее время никем не поддерживается и не развивается. В итоговую сравнительную таблицу данный Honeypot не попал.

5. *Formidable Honeypot.* Является дополнением для системы управления контентом Wordpress. Предназначен для защиты от спама форм обратной связи с пользователями сайта. Для работы с данным расширением необходимо установить еще одно расширение для работы с формами обратной связи – «Formidable Forms» и установить саму Formidable Honeypot на систему управления сайтом WordPress. После чего, как заверяют авторы, все формы обратной связи будут надежно защищены от спама.

6. *Blackhole for Bad Bots.* Расширение для системы управления контентом Wordpress. Предназначен для защиты сайта от зловредных ботов. Расширение работает весьма просто: для начала в файле robots.txt указываются те директории и страницы сайта, на которые доступ запрещен. Далее если же бот перейдет хотя бы по одной запрещенной ссылке, то данный бот более не сможет работать с данным сайтом. Однако при использовании различных анонимайзеров и средств VPN/Proxy данное ограничение можно обойти.

7. *Wordpot.* Представляет собой ловушку для системы управления контентом WordPress.

8. *HoneyThing.* Представляет из себя Honeypot интернета вещей. Поддерживает такие протоколы, как HTTP и CWMP. В настоящее время проект не поддерживается и не развивается, к тому же отсутствует внятная и понятная документация по установке и настройке. Последний комит был сделан 5 лет назад. В ходе анализа данной ловушки авторы пришли к выводу, что код проекта придется переписывать заново.

Авторами статьи было принято решение не включать данный Honeypot в итоговую таблицу.

9. *Dionaea.* Это Honeypot-система была разработана в рамках проекта Google Summer of Code в далеком 2009 году. Dionaea имеет модульную архитектуру, Python используется для эмуляции сетевых протоколов.

10. *Miniprint.* Данный Honeypot эмулирует работу сервера печати.

11. *Honeypot-ft.* Данный FTP Honeypot предлагает полную поддержку таких протоколов, как ftp и ftps. Данная ловушка позволяет производить полный мониторинг действий злоумышленника, начиная от несанкционированных попыток авторизации и заканчивая возможностью просмотра загруженных злоумышленником файлов для каждого сеанса FTP/FTPS.

В рамках исследования производилась установка и настройка, а также атака на каждую Honeypot-систему, по результатам которых были даны качественные показатели по описанным выше критериям. Оценки приведены в таблице 2.

Таблица 2 – Сравнение наиболее популярных систем-приманок

Имя приманки	Процесс установки и настройки	Процесс использования и поддержки	Уровень протоколирования	Уровень имитации	Уровень риска	Протокол(ы), сервис(ы), приложения
Cowrie	Средний	Средний	Высокий	Средний в режиме shell Высокий в режиме proxy	Низкий в режиме shell Высокий в режиме proxy	ssh, telnet
Kippo	Низкий	Средний	Высокий	Средний	Низкий	ssh, telnet
Glastopf	Высокий	Средний	Средний	Высокий	Низкий	Web-application
Formidable Honeypot	Низкий	Простой	Низкий	Высокий	Высокий	Расширение для системы управления контентом Wordpress

Продолжение таблицы 2

Blackhole for Bad Bots	Низкий	Простой	Низкий	Средний	Средний	Расширение для системы управления контентом Wordpress
Wordpot	Низкий	Простой	Низкий	Высокий	Низкий при грамотной настройке	80, 443 Web-ресурс
Dionaea	Сложный	Средний	Высокий	Высокий	Низкий	http, ftp, sip, smb, tftp, upnp, pptp, mssql, mqtt, memcache, epmap, blackhole
Miniprint	Простой	Простой	Высокий	Средний	Низкий	Сетевой принтер
Honeypot-ftp	Средний	Средний	Высокий	Высокий	Средний	ftp/ftps
telnet-iot-honeypot	Низкий	Низкий	Высокий	Низкий	Низкий	telnet

Сравнительный анализ методом анализа иерархий представлен на рисунке 5. Результатом приведенного сравнения стали 10 Honeypot-систем с качественными оценками критериев. Переводя качественные оценки в количественные и сравнивая системы между собой методом иерархий, выбирается наиболее оптимальная Honeypot-система. Результаты анализа приведены в таблице 3.

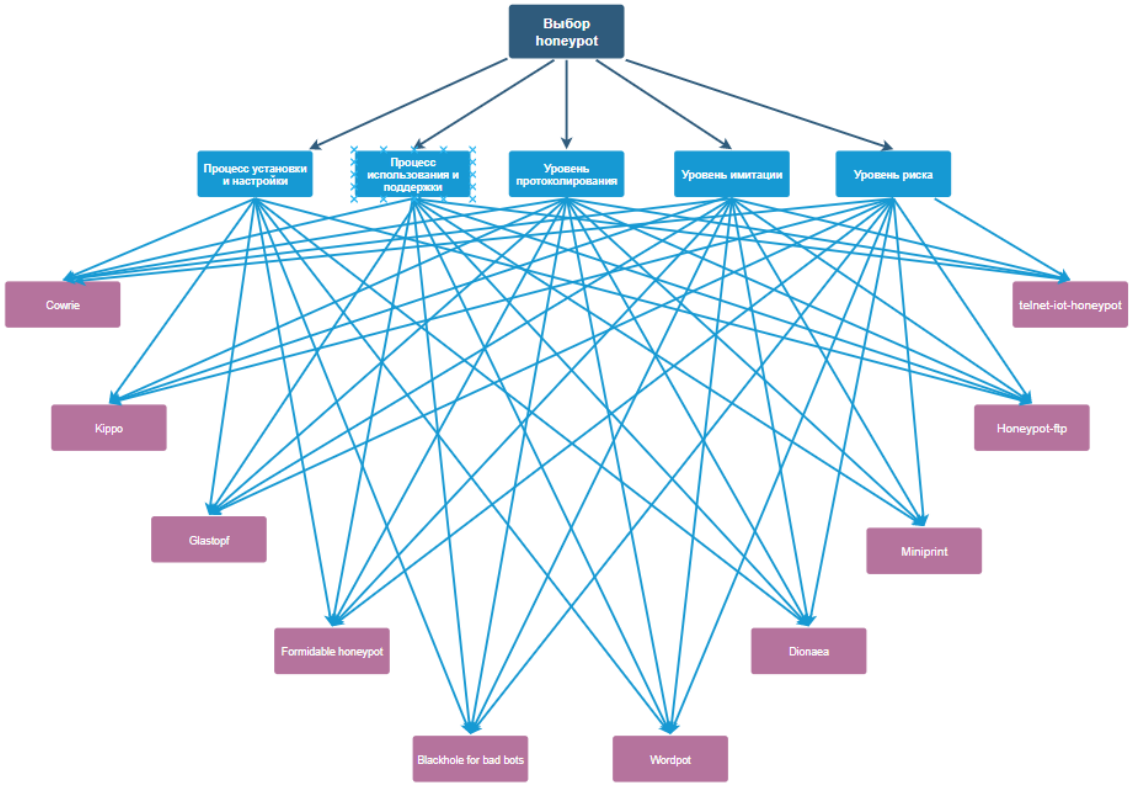


Рисунок 5 – Иерархия целей

В результате проведённого анализа нами определены наиболее оптимальные системы – Cowrie, Kippo и Glastopf. Далее с использованием системы Cowrie и дополнительного аппаратного обеспечения проведем разработку deception-комплекса для IoT-устройств.

Таблица 3 – Определение глобального приоритета альтернатив

	Процесс установки и настрой-ки	Процесс использова-ния и под-держки	Уровень протоколи-рования	Уровень имитации	Уровень риска взлома	Оценка
Норм. вектор приоритетов	0,0384	0,0717	0,2518	0,2164	0,4215	
Cowrie	0,042	0,041	0,2064	0,2729	0,043	**0,13370408**
Kippo	0,026	0,068	0,1588	0,1998	0,218	**0,18098356**
Glastopf	0,022	0,2448	0,0884	0,0635	0,223	**0,14839198**
Formidable Honeypot	0,256	0,1155	0,0564	0,071	0,019	0,05568617
Blackhole for Bad Bots	0,223	0,0877	0,1449	0,0907	0,03	0,08360959
Wordpot	0,054	0,1998	0,0217	0,1199	0,048	0,06804168
Dionaea	0,052	0,012	0,2945	0,0882	0,079	0,12939728
Miniprint	0,136	0,109	0,067	0,0502	0,0997	0,08279513
Honeypot-ftp	0,1	0,054	0,035	0,0365	0,1263	0,07765885
telnet-iot-honeypot	0,084	0,065	0,056	0,01272	0,111	0,0713702

Адаптация ловушек под уровень IoT. Адаптации cowrie под raspbery pi в режиме shell.

1. Установка зависимостей:

```
$ sudo apt-get install git python-virtualenv libssl-dev libffi-dev build-essential libpython3-dev python3-minimal authbind virtualenv –y
```

2. Создание пользователя cowrie:

```
$ sudo adduser --disabled-password cowrie
$ sudo su - cowrie
```

3. Скачиваем исходный код проекта с репозитория github:

```
$ git clone http://github.com/cowrie/cowrie
$ cd cowrie
```

4. Установка и активация виртуального окружения:

```
$ virtualenv --python=python3 cowrie-env
$ source cowrie-env/bin/activate
$ pip install --upgrade pip
$ pip install --upgrade -r requirements.txt
```

5. Конфигурация cowrie:

Произведем настройку cowrie

```
$ nano etc/cowrie.cfg.dist
```

Изменим имя хоста с svr04 на raspberry. Пример показан на рисунке 6.

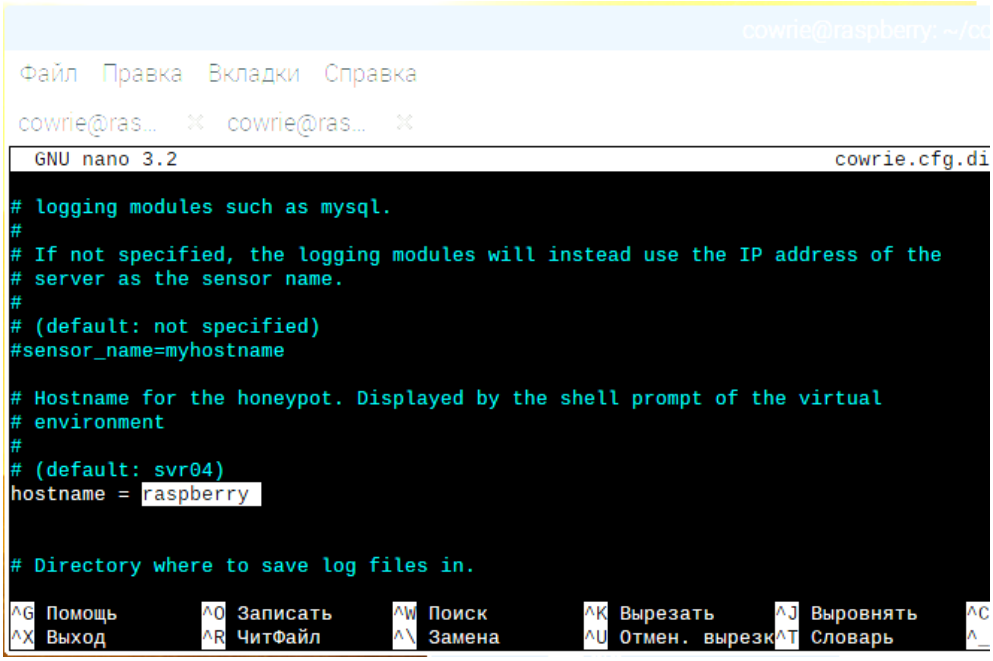


Рисунок 6 – Настройка cowrie под raspberry pi

Далее создадим файл userdb.txt в директории etc. В данном файле будут находиться пользовательские учетные данные, по которым злоумышленник, перебирая комбинации логин/пароль, будет производить попытки подключения к Honeypot.

```
$ touch etc/userdb.txt
$ nano etc/userdb.txt
```

В данном файле внесем одну запись, благодаря которой злоумышленник сможет успешно подключаться к cowrie практически с паролем от пользователя pi. Пример показан на рисунке 7.

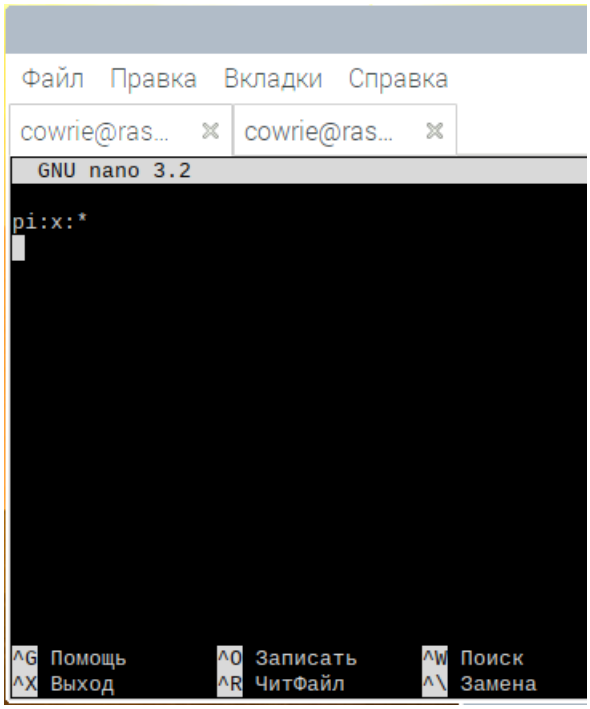


Рисунок 7 – Создание записи в файле userdb.txt

6. Для запуска достаточно выполнить следующую команду:

```
$ bin/cowrie start
```

7. Настройка порта 22.

Разработчики предусмотрели возможность настраивания cowrie таким образом, чтобы злоумышленники производили атаки на порт № 22, так как этот порт является стандартным портом ssh. Существуют 3 варианта установки cowrie на порт № 22, рассмотрим их:

7.1. Проброс портов через утилиту iptables. Для этого от имени пользователя root достаточно выполнить команду

```
$ sudo iptables -t nat -A PREROUTING -p tcp --dport 22 -j REDIRECT
--to-port 2222
```

После выполнения данной команды злоумышленник сможет подключаться к ловушке на порт № 22.

7.2. С помощью программы authbind. Для этого надо выполнить следующие команды:

```
$ sudo apt-get install authbind
$ sudo touch /etc/authbind/byport/22
$ sudo chown cowrie:cowrie /etc/authbind/byport/22
$ sudo chmod 770 /etc/authbind/byport/22
```

Далее в файле cowrie.cfg в настройках ssh надо поменять поле `listen_endpoints = tcp: 2222 : interface= 0.0 .0.0` на `listen_endpoints = tcp: 22 : interface= 0.0 .0.0`

7.3. Или использовать Setcap для того, чтобы дать возможность интерпретатору Python слушать порты, порядковый номер которых меньше, чем 1024.

В результате при подобной конфигурации при попытках подключения злоумышленника к cowrie будет производиться имитация подключения к устройству raspberry pi. А со стороны специалистов по информационной безопасности будут доступны сведения об активности злоумышленника (рис. 8 и 9).

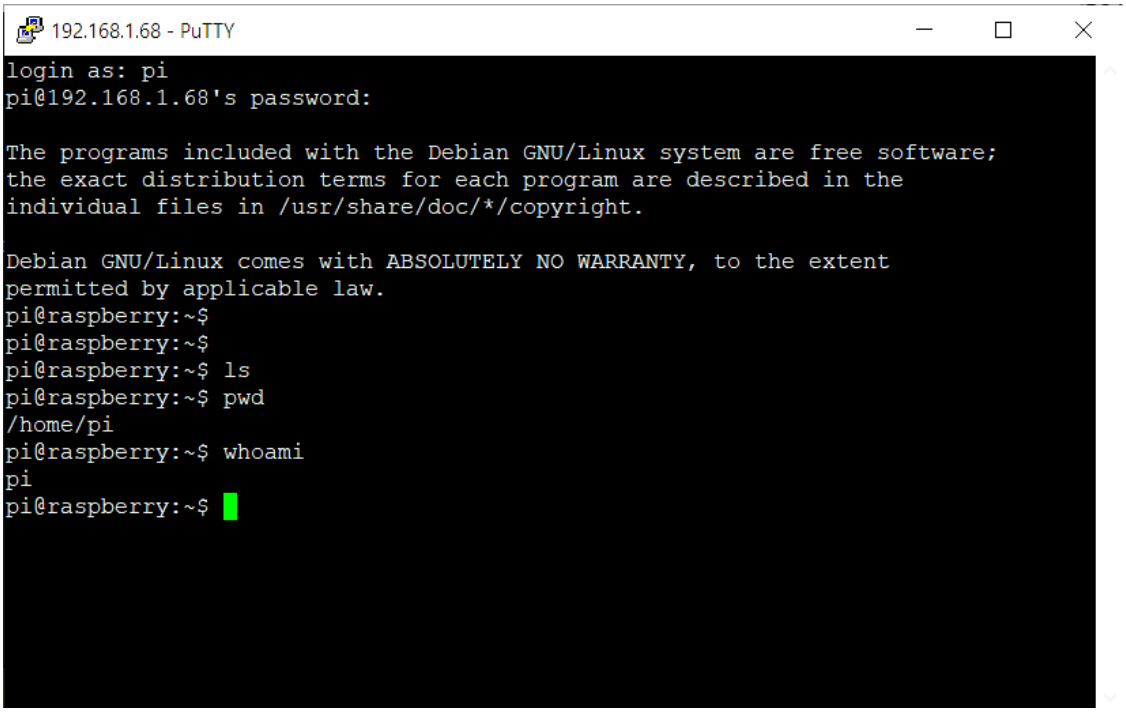


Рисунок 8 – Пример успешного подключения злоумышленника к cowrie

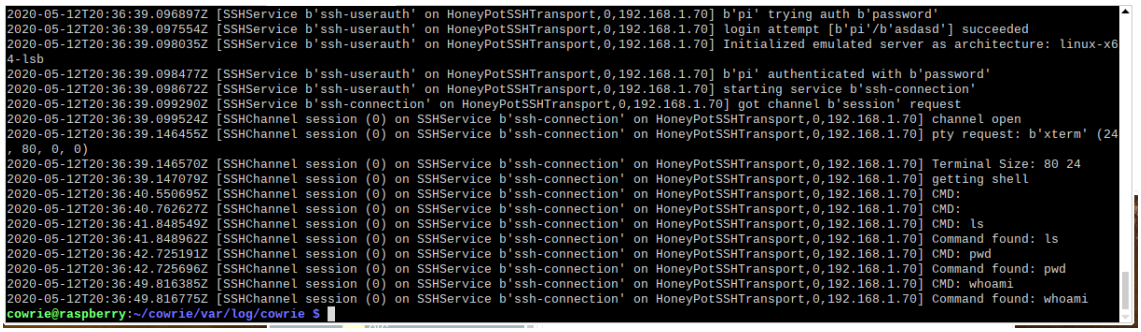


Рисунок 9 – Информационное окно cowrie при успешной компрометации

Выводы. Существующие на данный момент времени подложные информационные системы весьма обширны и выполняют свои функции. Эти системы вполне реально использовать и как монорешения, устанавливая на виртуальных частных серверах и внедряя в существующую сеть. С помощью такого класса устройств можно разворачивать целые сети, состоящие целиком и полностью из подложных устройств.

Однако в настоящее время не существует honeypot-системы, которая работала бы достаточно приемлемо на уровне устройств интернета вещей. Они имеют недостатки, что затрудняет их применение в практических целях. В большинстве своем данные ловушки морально устарели и требуют существенных доработок и модернизации. Например, упомянутый ранее в статье HoneyThing нуждается как в документации по установке и использованию, так и существенной модернизации своих функций. А telnet-iot-honeypot, несмотря на предоставляемые возможности для регистрации действий злоумышленника, очень легко раскрывается, так как отсутствует полноценная среда для взаимодействия злоумышленника с системой.

Сейчас очень важно разработать honeypot-систему интернета вещей, так как угрозы в этой области становятся все более опасными и массовыми. Это связано с тем, что безопасность интернета вещей на данный момент времени пребывает на низком уровне. С целью повышения уровня безопасности и уменьшения рисков киберугроз целесообразно развернуть honeypot, который будет производить сбор данных действий злоумышленников. На основе этих данных можно выявлять актуальные угрозы в сегменте интернета вещей и обеспечить защиту данных устройств своевременным выпуском обновлений системного и прикладного ПО.

Есть частные случаи проектирования отдельных honeypot-сетей с использованием реального и дорогостоящего оборудования [7], что значительно увеличивает обнаружение новой уязвимости и/или нового вектора атаки на устройства интернета вещей. Для проектирования такого рода сетей-приманок достаточно эмулирования реальных физических устройств. С появлением новых IoT honeypot стоимость выявления новой уязвимости или нового вектора атаки значительно снизится.

Для компаний, специализирующихся на разработке антивирусного ПО, актуально выявление новых уязвимостей и новых векторов атак на устройства интернета вещей. Единственно верной и действенной мерой для этого будет создание устройств приманок на уровне IoT.

Библиографический список

1. Алейнов Ю. В. Обнаружение атак направленного типа в компьютерных сетях при помощи ложных сетевых объектов / Ю. В. Алейнов // Информационное противодействие угрозам терроризма. – 2015. – № 24. – С. 52–57.

2. Врагова Е. В. Методы и средства защиты от botnet's (зомби-сетей) / Е. В. Врагова, Л. И. Воронова // Телекоммуникации и информационные технологии. – 2018. – Т. 5, № 1. – С. 112–116.

3. Куцовол Т. С. Технология honeypot как превентивный метод от сетевых атак / Т. С. Куцовол, И. С. Павлов // Меридиан. – 2019. – № 14 (32). – С. 45–47.

4. Хусни. Спектральный анализ трафика сети «honeypot» / Хусни // Труды СПИИРАН. – 2008. – № 7. – С. 177–180.

5. Как настроить собственный honeypot. – Режим доступа: https://www.anti-malware.ru/practice/solutions/How-To-Setup-Your-Own-Honeypot, свободный. – Заглавие с экрана. – Яз. рус.

6. Киберкомандование США провело операцию против одного из крупнейших ботнетов Trickbot https://www.securitylab.ru/news/512912.php, свободный. – Заглавие с экрана. – Яз. рус.

7. Невыразимо привлекателен: как мы создали ханипот, который нельзя разоблачить. – Режим доступа: https://habr.com/ru/company/trendmicro/blog/490406/, свободный. – Заглавие с экрана. – Яз. рус.

8. Обзор распределенных платформ для имитации инфраструктуры. Distributed Deception Platform. – Режим доступа: https://ko.com.ua/obzor_raspredelyonnyh_platform_dlya_imitacii_infrastruktury_distributed_deception_platform_ddp_127165, свободный. – Заглавие с экрана. – Яз. рус.

9. Обман злоумышленников с помощью ловушек TrapX DeceptionGrid. – Режим доступа: https://www.anti-malware.ru/practice/methods/TrapX-DeceptionGrid, свободный. – Заглавие с экрана. – Яз. рус.

10. Обзор рынка платформ для создания распределенной инфраструктуры ложных целей (Distributed Deception Platform). – Режим доступа: https://www.anti-malware.ru/analytics/Market_Analysis/Distributed-Deception-Platform, свободный. – Заглавие с экрана. – Яз. Рус

11. Очередько А. Р. Исследование siem-систем на основе анализа механизмов выявления кибератак / А. Р. Очередько, В. С. Герасименко, М. М. Путято, А. С. Макарян // Вестник Адыгейского государственного университета. Серия 4: Естественно-математические и технические науки. – 2020. – С. 25–31. – Режим доступа: http://vestnik.adygnet.ru/files/2020.2/6312/25-31.pdf, свободный. – Заглавие с экрана. – Яз. рус.

12. Сравнение технологий имитации и ловушек. – Режим доступа: https://roi4cio.com/produkty/sravnit/?template=28&p%5B1724%5D=1724&p%5B1646%5D=1646&p%5B1640%5D=1640&p%5B1638%5D=1638&p%5B1648%5D=1648&p%5B1650%5D=1650&p%5B1652%5D=1652&p%5B1654%5D=1654&p%5B1656%5D=1656, свободный. – Заглавие с экрана. – Яз. рус.

13. Технология honeypot. Часть 1: Назначение honeypot. – Режим доступа: https://www.securitylab.ru/analytics/275420.php, свободный. – Заглавие с экрана. – Яз. рус.

14. Технология honeypot. Часть 2: Классификация honeypot. – Режим доступа: https://www.securitylab.ru/analytics/275775.php, свободный. – Заглавие с экрана. – Яз. рус.

15. Угрозы интернета вещей и возможные методы защиты. – Режим доступа: https://os.kaspersky.ru/2019/03/13/ugrozy-interneta-veshhey-i-vozmozhnye-me/, свободный. – Заглавие с экрана. – Яз. рус.

16. Honeynet Project: ловушка для хакера. – Режим доступа: http://citforum.ru/security/internet/honeynet/, свободный. – Заглавие с экрана. – Яз. рус.

References

1. Aleynov Yu. V. Obnaruzhenie atak napravlennogo tipa v kompyuuternykh setyakh pri pomoshchi lozhnykh setevykh obektov. [Intrusion detection directions of the in-computer networks using false network objects]. *Informatsionnoe protivodeystvie ugrozam terrorizma* [Information counteraction to threats of terrorism], 2015, no. 24, pp. 52–57.

2. Vragova E. V., Voronova L. I. Metody i sredstva zashchity ot botnet's (zombi-setey) [Methods and means of protection from botnets (zombie-networks)]. *Telekommunikatsii i informatsionnye tekhnologii* [Telecommunications and information technology], 2018, vol. 5, no. 1, pp. 112–116.

3. Kutsovol T. S., Pavlov I. S. Tekhnologiya honeypot kak preventivnyy metod ot setevykh atak. [Honeypot technology as a preventive method against network attacks]. *Meridian* [Meridian], 2019, no. 14 (32), pp. 45–47.

4. Khusni. Spektralnyy analiz trafika seti «honeypot» [Spektral analysis of honeypot network traffic]. *Trudy SPIIRAN* [SPIIRAS Proceedings], 2008, no. 7, pp. 177–180

5. *Kak nastroit sobstvennyy honeypot* [How to set up your own honeypot]. Available at: https://www.anti-malware.ru/practice/solutions/How-To-Setup-Your-Own-Honeypot

6. Kiberkomandovanie SShA provelo operatsiyu protiv odnogo iz krupneyshikh botnetov Trickbot [The US cyber command conducted an operation against one of the largest botnets Trickbot]. Available at: https://www.securitylab.ru/news/512912.php

7. Nevyrazimo privlekatelen: kak my sozdali hanipot, kotoryy nelzya razoblachit [Unspeakably attractive: how we created a honeypot that can't be exposed]. Available at: https://habr.com/ru/company/trendmicro/blog/490406/

8. Obzor raspredelennykh platform dlya imitatsii infrastruktury. Distributed Deception Platform [Overview of distributed platforms for infrastructure simulation. Distributed Deception Platform]. Available at: https://ko.com.ua/obzor_raspredelyonnyh_platform_dlya_imitacii_infrastruktury_distributed_deception_platform_ddp_127165

9. Obman zloumyshlennikov s pomoshchyu lovushek TrapX DeceptionGrid [Deception of intruders using TrapX DeceptionGrid traps]. Available at: https://www.anti-malware.ru/practice/methods/TrapX-DeceptionGrid

10. *Obzor rynka platform dlya sozdaniya raspredelennoy infrastruktury lozhnykh tseley (Distributed Deception Platform)* [Market overview of platforms for creating distributed infrastructure for false goals (Distributed Deception Platform)]. Available at: https://www.anti-malware.ru/analytics/Market_Analysis/Distributed-Deception-Platform

11. Ochelyko A. R., Gerasimenko V. S., Putyato M. M., Makaryan A. S. Issledovanie siem-sistem na osnove analiza mekhanizmov vyyavleniya kiberatak [Research of siem-systems based on the analysis of mechanisms for detecting cyberattacks]. *Vestnik Adygeyskogo gosudarstvennogo universiteta. Seriya 4: Yestestvenno-matematicheskie i tekhnicheskie nauki* [Bulletin of the Adygea State University. Series 4: Natural-mathematical and technical sciences], 2020, pp. 25–31. Available at: http://vestnik.adygnet.ru/files/2020.2/6312/25-31.pdf

12. *Sravnenie tekhnologiy imitatsii i lovushek* [Comparison of simulation technologies and traps]. Available at: https://roi4cio.com/produkty/sravnit/?template=28&p%5B1724%5D=1724&p%5B1646%5D=1646&p%5B1640%5D=1640&p%5B1638%5D=1638&p%5B1648%5D=1648&p%5B1650%5D=1650&p%5B1652%5D=1652&p%5B1654%5D=1654&p%5B1656%5D=1656

13. *Tekhnologiya honeypot. Chast 1: Naznachenie honeypot* [Honeypot technology. Part 1: The purpose of the honeypot]. Available at: https://www.securitylab.ru/analytics/275420.php

14. *Tekhnologiya honeypot. Chast 2: Klassifikatsiya honeypot* [Honeypot technology. Part 2: Honeypot classification]. Available at: https://www.securitylab.ru/analytics/275775.php

15. *Ugrozy interneta veshchey i vozmozhnye metody zashchity* [Threats to the Internet of things and possible protection methods]. Available at: https://os.kaspersky.ru/2019/03/13/ugrozy-interneta-veshhey-i-vozmozhnye-me/

16. *Honeynet Project: lovushka dlya hakera* [Honeynet Project: a hacker's trap]. Available at: http://citforum.ru/security/internet/honeynet/