УДК 004.056.5

Д. С. Шанкин[1]*

Система автоматизированного реагирования на инциденты информационной безопасности категории фишинговых атак

[1] Национальный исследовательский ядерный университет «МИФИ», г. Москва, Российская Федерация
* e-mail: shankindanila0@yandex.ru

Аннотация. В статье рассмотрена реализация системы, позволяющей в автоматизированном режиме производить процесс реагирования на инциденты информационной безопасности категории фишинговых атак, а также проводить детальную аналитику подозрительных писем. Доступность инфраструктуры обеспечивается за счет снижения метрик времени реагирования, а также за счет реализации плана реагирования, разработанного в соответствии с существующими методологиями. Такая система сопровождает инцидент на всех этапах его жизненного цикла. Областью применения разработанной системы является обеспечение информационной безопасности корпоративных инфраструктур, в частности, конечных узлов инфраструктуры с внедренным почтовым решением. В результате анализа множественных сценариев запуска представленная система демонстрировала значительное улучшение в оперативности выполнения полного цикла реагирования на инцидент, исчисленного в тройном увеличении скорости.

Ключевые слова: фишинг, IRP, реагирование, индикатор компрометации

D. S. Shankin[1]*

System of secure remote access to operating systems of Linux families

[1] National Research Nuclear University MEPhI, Moscow, Russian Federation
* e-mail: shankindanila0@yandex.ru

Abstract. The article considers the implementation of a system that allows automated response to information security incidents in the category of phishing attacks, as well as detailed analytics of suspicious emails. Infrastructure availability is ensured by reducing response time metrics, as well as by implementing a response plan developed in accordance with existing methodologies. Such a system accompanies an incident at all stages of its lifecycle. The application area of the developed system is ensuring information security of corporate infrastructures, in particular, the end nodes of the infrastructure with the implemented email solution. As a result of analyzing multiple launch scenarios, the presented system demonstrated a significant improvement in the efficiency of performing the full cycle of incident response, calculated in triple speed.

Keywords: phishing, IRP, response, compromise indicator

Введение

Любая инфраструктура информационной безопасности не может гарантировать отсутствие вторжений или других злонамеренных действий. Когда происходят инциденты компьютерной безопасности, организациям крайне важно иметь эффек-

тивный способ определить, что что-то произошло, и принять меры реагирования. Скорость, с которой организация может распознать, проанализировать и отреагировать на инцидент, ограничит ущерб и снизит стоимость восстановления. Это основа возможностей управления инцидентами в организации [1].

Управление инцидентами требует от организаций создания процессов обнаружения, анализа, реагирования и извлечения уроков из инцидентов, которые угрожают конфиденциальности, доступности и целостности критически важных систем и данных. Эти функции поддерживаются процессами связи, координации, документирования и отслеживания деталей инцидентов и действий по реагированию.

Инциденты информационной безопасности влекут за собой остановку бизнес-процессов, потерю прибыли, а также репутации. Для снижения рисков таких потерь во время инцидентов, собираются команды для экстренного реагирования и выстраивания полного цикла процессов работы компании во время инцидента, которые называются IRT (англ. Incident Response Team) [2]. Главной целью данных команд является управление инцидентами, выстраивание жизненного цикла инцидента.

Для уменьшения времени реагирования соответствующие команды применяют концепцию автоматизации основных процессов, применимых в рамках работы с инцидентом. Однако существующие решения в индустрии и научной среде не обеспечивают достаточный уровень сопровождения инцидента на всех этапах его жизненного цикла.

По данным команды IRT компании Kaspersky, 37 % всех происходящих инцидентов в компании – это инциденты, связанные с фишинговыми рассылками. По экспертной оценке, аналитику требуется около 40 минут на проведение полного ручного исследования и реагирования на вредоносное письмо, что в рамках инцидента может стать критическим фактором.

В индустрии присутствуют решения, которые предоставляют некоторые средства автоматизации при реагировании на фишинговые атаки. Например, решение с открытым исходным кодом «ThePhish» [3]. Однако данный инструмент не позволяет проводить непосредственно процесс реагирования и не предоставляет возможности экспортировать вердикт для импорта в систему ведения инцидентов.

Также рассмотрим разработку, показанную в рамках научной работы под названием PhiGARo [4]. Данное решение обладает особенностями, которые ограничивают его применимость. Например, при отслеживании действий потенциально скомпрометированных пользователей, учитываются только переходы по прикрепленным ссылкам и ответные письма на фишинговые домены, когда в современных условиях существует множество других способов доставки полезной нагрузки, например, прикрепленные файлы, запускающие зловредный код при их открытии. Более того, обнаружение фишинговых писем базируется на внедрении в инфраструктуру приманок для злоумышленников, в виде поддельных почтовых доменов-триггеров [5], которые оповещают сотрудников безопасности о наличии нелегитимной рассылки. Такая независимость от пользователей при-

вносит некоторые недостатки, связанные с информационной безопасностью, которые возникают от внедрения приманок.

Постановка задачи

Целью настоящей работы является обеспечение доступности информационной инфраструктуры предприятия посредством системы, позволяющей в автоматизированном режиме реагировать на инциденты информационной безопасности категории фишинговых атак

В ходе работы была проведена ее декомпозиция на следующие задачи:

1) исследование предметной области;
2) проектирование системы автоматизированного реагирования;
3) программная реализация предложенного решения;
4) проведение автоматизированного тестирования.

Предлагаемое решение

Проектирование рассматриваемого решения требует детального рассмотрения вопроса жизненного цикла инцидента. В индустрии принято выделять следующие этапы работы команды реагирования при реализации недопустимого события [6–7]:

1) подготовка – формирование стратегии и механизмов реагирования на инциденты, разработка планов реагирования, тренировка персонала;
2) детектирование – поиск признаков инцидентов с использованием инструментов мониторинга и анализа трафика;
3) анализ – выявление сущности и характера инцидента, определение используемых методов и тактик;
4) локализация/сдерживание – ограничение распространения инцидента и предотвращение дополнительного ущерба;
5) устранение – удаление угрозы, восстановление пораженных ресурсов;
6) восстановление – возврат системы в нормальное состояние;
7) пост-инцидентные действия – анализ произошедшего, вынесение уроков для предотвращения подобного в будущем.

Разработанный план реагирования (рис. 1) полностью соответствует описанной системе жизненного цикла инцидента, а также учитывает особенности [8], а также типичные признаки, присущие рассматриваемым социально-техническим зловредным письмам, например [9]:

1) содержание самого письма;
2) выставленные заголовки безопасности и аутентификации;
3) приложенные зловредные файлы;
4) приложенные фишинговые ссылки;

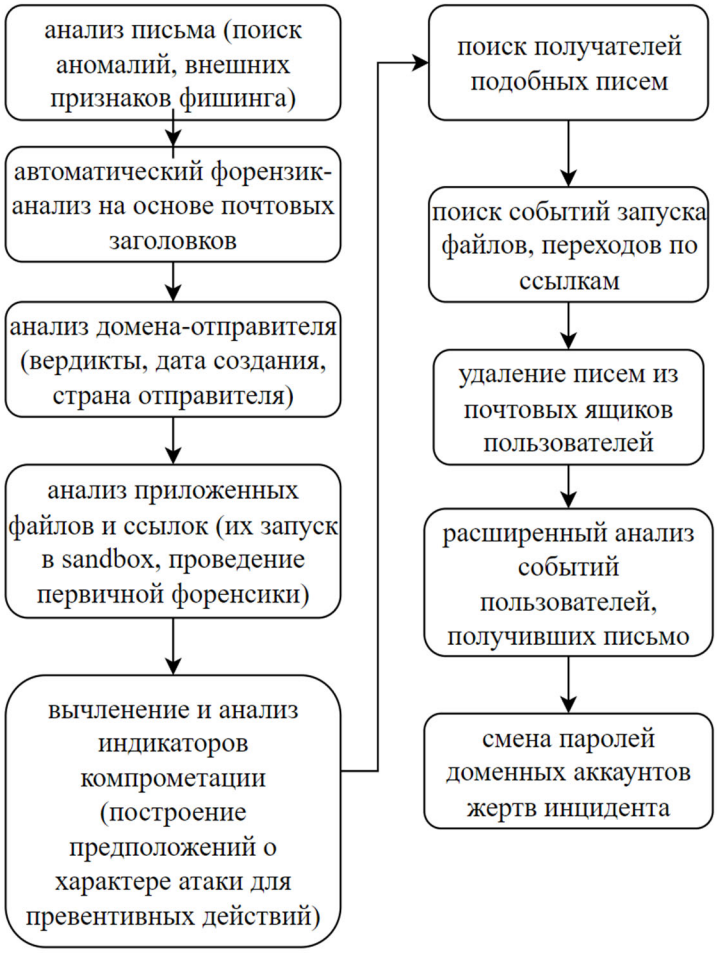

Рис. 1. Разработанный план реагирования

В процессе проектирования предложенной системы было рассмотрено многочисленное множество механизмов, решений и вспомогательных средств защиты информации с тем, чтобы обеспечить полную агрегацию информации о входящих сообщениях в единой консоли управления. В ходе этого процесса были определены важнейшие связи между выделенными объектами (рис. 2) с целью гарантировать конфиденциальность, доступность, подотчетность и непрерывность разрабатываемого решения.

Конфиденциальность данных обеспечивается благодаря применению специализированного инструмента для безопасного хранения секретов, такого как Hashicorp Vault [10], где реализована ролевая модель доступа к хранимым API ключам. Для обеспечения доступности системы используется Docker контейнер [11] с встроенной функцией мониторинга состояния, позволяющей автоматически перезапускать контейнер в случае нестабильности работы. Подотчетность обеспечивается специально разработанным модулем, ответственным за регистрацию информации о запросах и сохранение данных в отдельные файлы. Непрерывность же обеспечивается за счет методик разработки DevOps [12] с внед-

ренными GitLab CI и GitLab Runner [13], которые вводятся в эксплуатацию автоматизировано с помощью безопасных Ansible сценариев автоматизации [14-16].

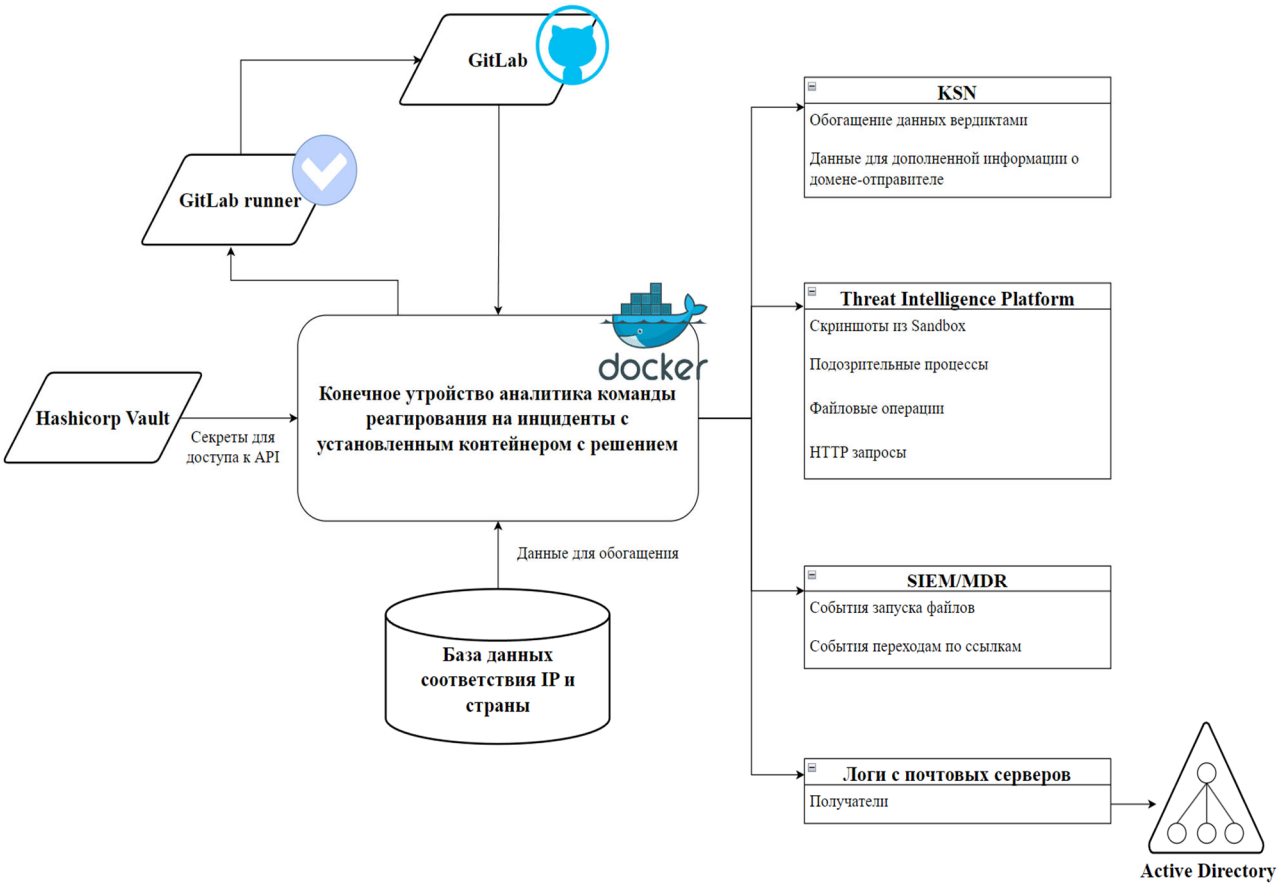

Рис. 2. Диаграмма взаимодействия объектов

Для программной реализации спроектированной системы был разработан специальный алгоритм (рис. 3), производящий в автоматизированном режиме реализацию всех пунктов плана реагирования на почтовые угрозы.

На первом этапе система принимает входящее электронное сообщение, анализирует его начальную информацию, такую как тема, дата и содержание. Это позволяет выделить основные атрибуты письма и определить его потенциально фишинговый характер.

Вторым этапом происходит работа модуля проверки заголовков аутентификации. Он позволяет эффективно фильтровать и идентифицировать подозрительные электронные письма, снижая риск попадания пользователей под атаки фишинга и предотвращая утечку конфиденциальной информации. Благодаря использованию стандартов аутентификации и проверки подлинности, модуль обеспечивает более высокий уровень безопасности электронной почты в организации.

Далее система направляет внимание на домен-отправитель, проводя анализ WHOIS данных для получения информации о домене. Дополнительно, осуществ-

ляется запрос к различным базам данных для оценки доверенности домена. Такой подход помогает выявить потенциально злонамеренные домены и повысить уровень доверия к отправителям.

Интеграция с логами почтового сервера позволяет системе извлекать список получателей письма, что обогащает контекст фишинговой атаки и может быть использовано для более детального анализа.

Следующий шаг включает интеграцию с системой централизованного мониторинга (SIEM/MDR), что обеспечивает возможность анализа действий пользователей в отношении приложенных файлов или переходов по фишинговым ссылкам. Таким образом, система способна отслеживать взаимодействие пользователей с потенциально опасными элементами письма.

Интеграция с корпоративным каталогом Active Directory предоставляет доступ к разнообразной информации о пользователях, например, дате последней смены пароля, UPN пользователя и другим данным, что дополнительно обогащает контекст анализа.

Следующие два модуля, отвечающие за анализ приложенных ссылок и файлов, работают по принципу сбора вердиктов, создания скриншотов и анализа HTTP запросов для ссылок, а также анализа запускаемых процессов, файловых и регистровых операций для файлов. Это позволяет выявлять вредоносные элементы и повышает эффективность обнаружения фишинговых атак.

На последнем этапе система формирует письма для участвующих в инциденте подразделений компании. Таким образом, система обеспечивает полный цикл обработки и реагирования на фишинговые инциденты, снижая риски для информационной безопасности компании.

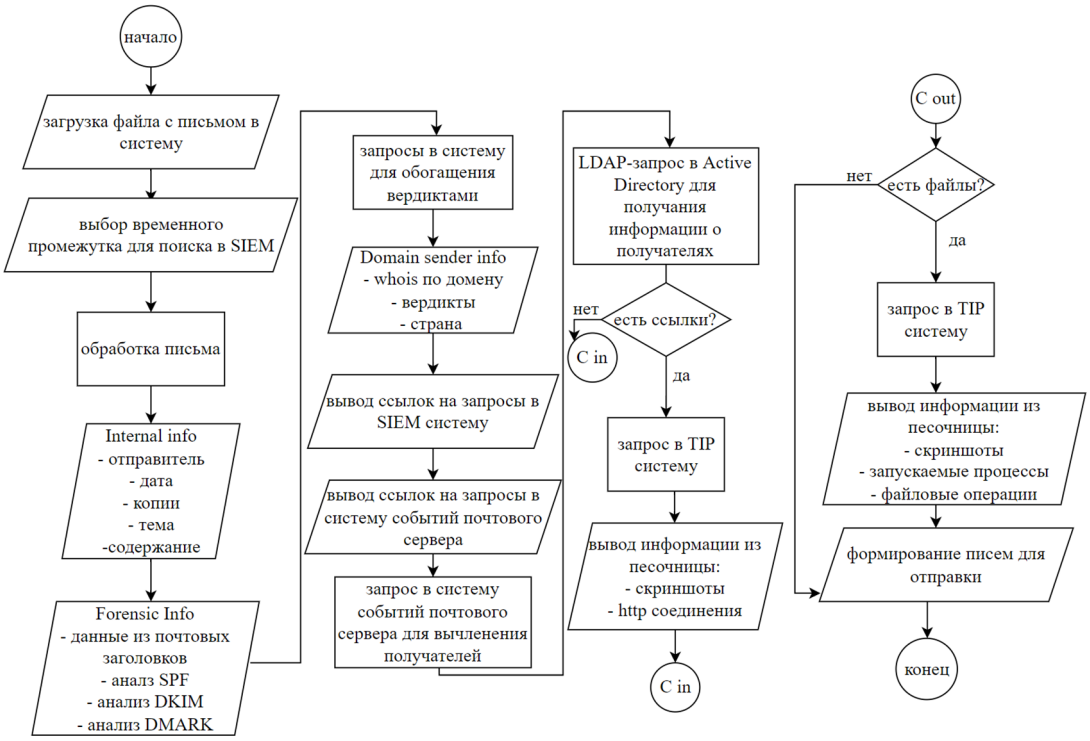

Рис. 3. Блок-схема разработанного алгоритма

По итогу работы алгоритма формируется отчет, содержащий в себе информацию о письме, его заголовках, отправителе, приложенных файлах и ссылках с вердиктами из различных систем, список и количество получателей с расширенной информацией из системы каталогов, гибкие ссылки, составленные в системы мониторинга для детального анализа событий журналов, связанных с данным инцидентом, а также набор сформированных писем в различные подразделения компании для распределения задач в рамках реагирования на инцидент.

Тестирование

В данной работе для демонстрации работоспособности и корректности предложенного решения проводилось модульное тестирование, а также ручное тестирование [17].

Модульное тестирование позволяет осуществить проверку отдельных модулей программного обеспечения с целью выявления и исправления потенциальных дефектов и ошибок в их функционировании. В данной работе модульные тесты, внедренные в процесс непрерывной поставки и тестирования CI/CD, тестируют корректность работы функций обработки информации, получаемой системой из API запросов, а также запросов в такие системы, как Active Directory.

Ручное тестирование заключалось в запуске решения на тестовом наборе искусственно сформированных писем для проверки корректности отображения каждого и модулей, а также на наборе писем, которые по итогу анализа были выделены как фишинговые.

Заключение

В рамках данной работы была предложена система автоматизированного реагирования на инциденты информационной безопасности категории фишинговых атак, которая была реализована и успешно себя продемонстрировала на этапе тестирования.

Теоретическая значимость работы заключается в проектировании плана реагирования, а также алгоритма его автоматизации, обеспечивающие полноту и достоверность данного впоследствии анализа вердикта.

Практическая значимость заключается в обеспечении безопасности при возникновении почтовых угроз, что позволяет снизить риск нанесения ущерба инфраструктуре предприятия путем снижения времени реагирования.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Милославская Н. Г., Сенаторов М. Ю., Толстой А. И. Управление инцидентами информационной безопасности и непрерывностью бизнеса. Учебное пособие. – 2012.

2. Ruefle R. et al. Computer security incident response team development and evolution // IEEE Security & Privacy. – 2014. – Т. 12. – №. 5. – С. 16-26.

3. ThePhish. – URL: https://github.com/emalderson/ThePhish (02.12.2023).

4. Husák M., Cegan J. PhiGARo: Automatic phishing detection and incident response framework // 2014 ninth international conference on availability, reliability and security. – IEEE, 2014. – С. 295-302.

5. Martinez C., Thorpe C. Analysis of spam: honeypot experiment // European Conference on Cyber Warfare and Security. – Academic Conferences International Limited, 2017. – С. 692-701.

6. Виноградова М. В., Грудинин Д. О. Разработка модели плана реагирования на инцидент и алгоритма оценки его эффективности для автоматизированных информационных систем // Актуальные проблемы прикладной математики, информатики и механики. – 2020. – С. 1811-1817.

7. Thompson E. C. Cybersecurity incident response: How to contain, eradicate, and recover from incidents. – Apress, 2018.

8. Ma L. et al. Detecting phishing emails using hybrid features // 2009 Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing. – IEEE, 2009. – С. 493-497.

9. Kumaraguru P. et al. Protecting people from phishing: the design and evaluation of an embedded training email system // Proceedings of the SIGCHI conference on Human factors in computing systems. – 2007. – С. 905-914.

10. Dykstra D., Altunay M., Teheran J. Secure command line solution for token-based authentication // EPJ Web of Conferences. – EDP Sciences, 2021. – Т. 251. – С. 02036.

11. Bui T. Analysis of docker security // arXiv preprint arXiv:1501.02967. – 2015.

12. Ebert C. et al. DevOps // IEEE software. – 2016. – Т. 33. – №. 3. – С. 94-100.

13. Singh C. et al. Comparison of different CI/CD tools integrated with cloud platform // 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence). – IEEE, 2019. – С. 7-12.

14. Басыня Е. А., Лукина М. С. Автоматизированная установка и конфигурирование серверных решений //Современные материалы, техника и технологии. – 2016. – №. 2 (5). – С. 21-26.

15. Юшманов А. А. Разработка комплексной охранной системы // Интерэкспо Гео-Сибирь. – 2020. – Т. 7. – №. 1. – С. 140-148.

16. Костромин Р. О. Сравнительный обзор средств управления конфигурациями ресурсов вычислительной среды функционирования цифровых двойников //Информационные и математические технологии в науке и управлении. – 2021. – №. 1 (21). – С. 132-145.

17. Basili V. R., Selby R. W. Comparing the effectiveness of software testing strategies //IEEE transactions on software engineering. – 1987. – №. 12. – С. 1278-1296.