Инженерный вестник Дона, №6 (2023)
ivdon.ru/ru/magazine/archive/n6y2023/8447

Анализ уязвимостей в системах безопасности данных

А.Р. Хромова[1], *Л.Э. Петросян*[1,2]

[1] *Московский государственный университет технологий и управления им. К.Г. Разумовского (Первый казачий университет)*
[2]*Московский институт радиотехники, электроники и автоматики*

Аннотация: Статья представляет собой обзорную работу, посвященную методам и технологиям, используемым при анализе уязвимостей в информационных системах. В статье описываются основные этапы проведения анализа уязвимостей, такие, как сбор информации о системе, сканирование системы на уязвимости и анализ результатов сканирования. Также рассматриваются методы защиты от уязвимостей, такие, как регулярное обновление программного обеспечения, проведение анализа уязвимостей и разработка стратегии безопасности данных.
Ключевые слова: анализ уязвимостей, безопасность данных, угрозы информационной безопасности, защита от атак, информационная безопасность, компьютерная безопасность, риск безопасности, уязвимость сети, система безопасности, защита.

Анализ уязвимостей в системах безопасности данных - это процесс оценки уровня безопасности информационной системы, с целью выявления уязвимостей и разработки мер для их устранения. Уязвимость - слабое место в системе, которое может быть использовано злоумышленниками для нарушения ее безопасности и целостности [1].

Целью анализа уязвимостей является обнаружение потенциальных уязвимостей в системе и разработка мер по их устранению или минимизации рисков. Для этого проводится исследование архитектуры системы, программного обеспечения, настроек безопасности и других факторов, которые могут повлиять на безопасность данных.

Определение уязвимостей — это процесс выявления слабостей в системе, которые могут быть использованы злоумышленниками для нарушения ее безопасности. Уязвимости могут возникать из-за ошибок в программном обеспечении, недостаточной защиты паролей, открытых портов и других факторов, что могут послужить для несанкционированного доступа к информации [2, 3]. Проведение анализа уязвимостей помогает выявить эти слабости и принять меры для их устранения.

Анализ уязвимостей является важным инструментом для обеспечения безопасности информации в системах. Он позволяет выявить потенциальные уязвимости и риски нарушения безопасности данных, которые могут быть использованы злоумышленниками для несанкционированного доступа, изменения или уничтожения информации [4, 5].

Одной из главных причин проведения анализа является обеспечение безопасности информации. Защита данных - важный фактор для бизнеса, государственных организаций, медицинских учреждений и других учреждений. Утечка конфиденциальной информации может привести к серьезным последствиям, включая потерю репутации, юридические проблемы и финансовые потери [3, 6].

Важной целью является также соответствие требованиям законодательства, включая нормы по защите персональных данных. Некоторые законы и стандарты, например, GDPR (General Data Protection Regulation) – это регламент Европейского союза (ЕС), определяющий порядок обработки персональных данных организациями. Если ваша организация продает товары или предоставляет услуги гражданам ЕС, или нанимает их на работу, вам необходимо соблюдать требования GDPR [1,6]. HIPAA (Health Insurance Portability And Accountability Act) – это акт (закон) о мобильности и подотчётности медицинского страхования. Был одобрен, чтобы модернизировать поток медицинской информации, предсказать, как личная информация, хранящаяся в медицинских учреждениях и медицинских страховых отраслях, должна быть защищена от мошенничества и краж, а также обращаться к ограничениям на медицинское страхование [6]. DSS (Decision Support System) – это информационная система, которая поддерживает деловую или организационную деятельность по принятию решений [5]. Эти законы и стандарты обязывают организации обеспечивать

высокий уровень безопасности информации, а также регулярно проверять их системы на наличие уязвимостей.

Помимо обеспечения безопасности информации и соответствия законодательству, анализ уязвимостей может также помочь предотвратить финансовые потери. Некоторые уязвимости могут быть использованы для кражи финансовой информации или нарушения работы системы, что может привести к прямым финансовым потерям для организации.

Информационные технологии помогают нам работать, общаться, и многое другое. Однако, с ростом количества данных, которые мы передаем и храним в сети, возрастает и риск их утечки и несанкционированного доступа к ним. В этом контексте, обеспечение безопасности данных становится одним из наиболее важных аспектов при использовании информационных технологий. Анализ уязвимостей в системах безопасности данных позволяет идентифицировать уязвимости в системе и предпринять меры по их устранению, что гарантирует сохранность данных и защиту от возможных кибератак [4].

Давайте рассмотрим некоторые из наиболее распространенных уязвимостей, которые могут быть обнаружены при проведении анализа. Наиболее распространенные уязвимости, которые могут быть использованы злоумышленниками для атак на системы, включают в себя SQL-инъекции (Structured Query Language), кросс-сайтовый скриптинг и уязвимости веб-приложений.

SQL-инъекция - уязвимость, которая позволяет атакующему использовать фрагмент вредоносного кода на языке структурированных запросов (SQL) для манипулирования базой данных и получения доступа к потенциально ценной информации [7]. SQL-инъекции являются одними из наиболее распространенных уязвимостей в системах управления базами данных. Эта уязвимость возникает из-за недостаточной валидации

пользовательского ввода, что может привести к тому, что злоумышленник выполняет вредоносный SQL-запрос, который позволит ему получить доступ к защищенной информации в базе данных. Ниже таблица №1 демонстрирует примеры вредоносных SQL-запросов и возможные последствия таких атак.

Таблица №1

Примеры вредоносных SQL-запросов

Тип атаки	Пример запроса	Последствия
SQL-инъекция через идентификатор	SELECT * FROM users WHERE id='1' OR 1=1;	Получение доступа к конфиденциальной информации из базы данных.
SQL-инъекция с вторжением в систему	SELECT * FROM users WHERE username = 'admin' AND password = 'password' OR '1' = '1'	Получение доступа к данным любого пользователя, не зная его логин и пароль.
SQL-инъекция через комментарий	SELECT * FROM users WHERE username = 'admin' --	Получение списка пользователей из базы данных, в том числе их логинов и паролей.

Кросс-сайтовый скриптинг XSS (Cross-Site Scripting) — это уязвимость, которая позволяет злоумышленникам внедрять вредоносный JavaScript-код на веб-страницы, просматриваемые пользователями. Это может позволить злоумышленнику получить доступ к конфиденциальным данным пользователя или даже выполнить действия от его имени. Таблица №2 ниже демонстрирует примеры вредоносного JavaScript-кода и возможные последствия таких атак.

Таблица №2

Примеры вредоносного JavaScript-кода

		Последствия
XSS атака	<script>alert('XSS')</script>	Позволяет злоумышленнику внедрить вредоносный код на страницу и получить доступ к личной информации пользователей.
Clickjacking (атака на манипуляцию кликами)	<iframe src="malicious.com/evil.html"> </iframe>	Злоумышленник может получить доступ к конфиденциальным данным и выполнить действия от имени пользователя, такие, как покупка товаров или выполнение транзакций.
CSRF атака	<img src="example.com/delete?id=1">	Может привести к несанкционированным действиям на стороне пользователя, таким, как перевод денежных средств на другой счет.

Уязвимости веб-приложений — это широкий класс уязвимостей, связанных с веб-приложениями, которые могут привести к утечке конфиденциальной информации или выполнению неавторизованных действий. Это может включать в себя уязвимости в аутентификации и

авторизации, вводе данных, контроле доступа, обработке входных данных, а также в различных компонентах веб-приложения, таких, как формы, скрипты, базы данных и т.д. [8, 9].

Примером уязвимости веб-приложений является SQL-инъекция, которая позволяет злоумышленнику выполнять произвольный SQL-код в базе данных, используемой веб-приложением. Это может привести к утечке конфиденциальной информации, такой, как пароли и данные пользователей, или к модификации и удалению данных в базе.

Анализ уязвимостей веб-приложений позволяет выявить и исправить подобные уязвимости, повышая безопасность веб-приложений и предотвращая утечку конфиденциальной информации или неавторизованные действия.

Конечная цель анализа уязвимостей в системах безопасности данных — это определение и устранение уязвимостей, которые могут привести к потере конфиденциальности, целостности или доступности данных [10]. Для достижения этой цели можно использовать нижеследующие этапы.

Первый этап - сбор информации о системе. На данном этапе проводится сбор информации о системе, которую необходимо проверить. Этот этап может включать в себя сбор информации о версии используемых операционных систем, веб-серверов, приложений и других компонентов системы. Важно получить максимально полную информацию о системе, чтобы понимать, какие именно уязвимости могут быть использованы злоумышленниками.

Второй этап - сканирование системы на уязвимости. На этом этапе проводится сканирование системы на наличие известных уязвимостей. Это может быть выполнено с помощью различных инструментов сканирования уязвимостей, которые будут искать уязвимости в системе, используя базы данных уязвимостей и другие средства. Также может быть проведено ручное

сканирование, включающее в себя тестирование системы с использованием различных техник.

Третий этап - анализ результатов сканирования. На этом этапе происходит анализ результатов сканирования. Все обнаруженные уязвимости должны быть проанализированы и оценены на основе их потенциальной угрозы и риска для системы. Это позволит определить, какие уязвимости требуют наивысшего приоритета для исправления, а также какие дополнительные меры могут быть предприняты для уменьшения рисков.
После анализа результатов сканирования можно составить отчет о найденных уязвимостях и рекомендациях по их исправлению.

Существует несколько мер, которые помогут уменьшить риски возникновения уязвимостей и снизить вероятность несанкционированного доступа к данным. Ниже приведены три основных шага, которые необходимо выполнять для обеспечения безопасности данных:

1. Обновление программного обеспечения: часто уязвимости появляются из-за устаревшего программного обеспечения. Поэтому важно регулярно обновлять все программные компоненты, включая операционные системы, приложения, библиотеки и фреймворки. Разработчики выпускают обновления, которые исправляют уязвимости в программном обеспечении, поэтому необходимо следить за обновлениями и устанавливать их вовремя.
2. Проведение регулярного анализа уязвимостей позволяет обнаружить возможные проблемы в системе безопасности данных и принять меры по их устранению. Анализ уязвимостей должен включать как автоматическое, так и ручное сканирование, чтобы обнаружить широкий спектр уязвимостей.
3. Разработка стратегии безопасности данных позволяет определить, какие данные являются наиболее ценными и важными для защиты,

какие угрозы могут возникнуть и какие меры защиты необходимо принять. Это включает в себя разработку политик доступа, шифрование данных, контроль доступа и многое другое.

Заключение

Анализ уязвимостей в системах безопасности данных является критически важным шагом для обеспечения безопасности в организации. Атаки на информационные системы могут привести к серьезным последствиям, таким, как утечка конфиденциальной информации, нарушение бизнес-процессов и финансовые потери.

Следуя процедурам анализа уязвимостей, организации могут выявить потенциальные уязвимости в своей системе безопасности данных и принять меры для их устранения. Кроме того, регулярное проведение анализа уязвимостей может помочь организациям поддерживать эффективную стратегию безопасности данных.

Однако, следует понимать, что анализ уязвимостей — это непрерывный процесс и его необходимо проводить регулярно, так как новые уязвимости могут появляться в любой момент. Поэтому важно разработать стратегию безопасности данных, которая будет включать в себя регулярное обновление программного обеспечения, проведение анализа уязвимостей и обучение сотрудников правилам безопасности.

Литература

1. Xarper A., Makkirnan D. Gray hat hacking: the ethical hacker's handbook. McGraw-Hill/Osborne 2005, Pp. 268 – 273.
2. Антипенко А. А. Защита веб-приложений - Издательство "БХВ-Петербург». 2012, 168 с.

3. Плёнкин А.П. Симметричное шифрование квантовыми ключами // Инженерный вестник Дона. 2016. №3. URL: ivdon.ru/ru/magazine/archive/n1y2016/3705.
4. Петров П.П. Сидоров С.С. Анализ и экстраполяция тренда отношения российского среднего класса к текущему политическому порядку // Научная мысль Кавказа. Междисциплинарный журнал. 2009. №3, 316-318 с.
5. Чепижны Д.В. Web-уязвимости и их эксплуатация. Издательство "БХВ-Петербург", 2014, 305 – 307 с.
6. Иванов И.И. Управление маркетинговыми исследованиями в регионе. Новочеркасск: НГТУ, 2004, 256 с.
7. Алгазали С.М.М, Айвазов В.Г., Кузнецова А.В. Совершенствование процесса поиска неэффективных SQL-запросов в СУБД Oracle // Инженерный вестник Дона. 2017. №4. URL: ivdon.ru/ru/magazine/search?search=Алгазали+С.М.М
8. Кобзарь А. С. Веб-безопасность - издательство "Ленанд", 2016, 94 – 99 с.
9. Sallivan B. Web Application Security, A Beginner's Guide (Beginner's Guide) McGraw Hill; 1st edition (November 24, 2011), Pp. 217 – 225.
10. Демиденко А.М. SQL-инъекции: методы атак и защиты. - ООО "Диалектика", 2015, 79 – 84 с.

References

1. Xarper A., Makkirnan D. Gray hat hacking: the ethical hacker's handbook.McGraw-Hill/Osborne 2005, Pp. 268 – 273.
2. Antipenko A. A. Zashhita veb-prilozhenij [Web application protection]. Izdatel`stvo "BXV-Peterburg") 2012, 168 p.

3. Pljonkin A.P. Inzhenernyj vestnik Dona, 2016, №3. URL: ivdon.ru/ru/magazine/archive/n1y2016/3705.
4. Petrov P.P. Sidorov S.S. Nauchnaya my`sl` Kavkaza. Mezhdisciplinarny`j zhurnal. 2009. №3, Pp.316-318.
5. Chepizhny` D.V. Web-uyazvimosti i ix e`kspluataciya [Web-vulnerabilities and their exploitation]. Izdatel`stvo "BXV-Peterburg", 2014, Pp. 305 – 317.
6. Ivanov I.I. Upravlenie marketingovy`mi issledovaniyami v regione [Management of marketing research in the region]. Novocherkassk: NGTU, 2004, 256 p.
7. Algazali S.M.M, Ajvazov V.G., Kuzneczova A.V. Sovershenstvovanie processa poiska nee`ffektivny`x SQL-zaprosov v SUBD Oracle // Inzhenerny`j vestnik Dona. 2017. №4. URL: ivdon.ru/ru/magazine/search?search=Algazali+S.M.M
8. Kobzar` A. S. Veb-bezopasnost` [Web-security]. Izdatel`stvo "Lenand", 2016, Pp. 94 – 99.
9. Sallivan B. Web Application Security, A Beginner's Guide (Beginner's Guide) McGraw Hill; 1st edition (November 24, 2011), Pp. 217 – 225.
10. Demidenko A.M. SQL-in``ekcii: metody` atak i zashhity` [SQL injection: attack and protection methods]. OOO "Dialektika", 2015, Pp. 79 – 84.