УДК 004.056.53

Д. А. Вислобоков, И. С. Шишкин, Е. С. Маленков[*]

ПРИМЕНЕНИЕ ТЕХНОЛОГИИ КИБЕРОБМАНА ДЛЯ ЗАЩИТЫ ФАЙЛОВЫХ СЕРВЕРОВ

Введение

Файловые серверы выполняют ключевую функцию в информационной инфраструктуре организации – они обеспечивают централизованное хранение, управление и обмен файловыми ресурсами: рабочими документами, резервными копиями, конфигурациями и учетными данными. Нарушение целостности, конфиденциальности или доступности этих серверов приводит к прямым коммерческим потерям, срыву бизнес-процессов и риску массовой компрометации пользовательских и внутренних данных [1].

В последние годы наблюдается высокая доля инцидентов, связанных с эксплуатацией уязвимостей в средствах управления и передачи файлов. Яркий пример – масштабные инциденты, связанные с уязвимостью в ПО для передачи файлов MOVEit, которые легли в основу значительной части уведомлений о нарушениях данных в исследовании по утечкам за 2023–2024 годы. Это подчеркивает уязвимость решений, используемых для обмена и хранения файлов [1].

Одновременно классические протоколы передачи файлов сохраняют широкое распространение. По оценкам, в сети остаются миллионы активных FTP-серверов, а уязвимости и неправильная конфигурация открытых сервисов продолжают делать такие ресурсы легкой целью для злоумышленников [2].

Применение технологии киберобмана для защиты файловых серверов

Технология киберобмана – это целенаправленное развертывание в сети ложных ресурсов, имитирующих реальные активы, в целях выявления и изучения вредоносной активности. Принцип работы

[*] Работа выполнена под руководством доктора технических наук, профессора кафедры «Информационные системы и защита информации» ФГБОУ ВО «ТГТУ» В. В. Алексеева.

основан на создании ложных целей – поддельные серверы, файлы, учетные записи и т.д., которые похожи на реальные. Ложные объекты размещают в таких местах сети и с такими правами доступа, чтобы они могли заинтересовать злоумышленника, но при этом не давали доступа к реальным данным. Любая попытка взаимодействия с ложным объектом рассматривается как индикатор компрометации. Поскольку легитимные пользователи не работают с этими целями, вероятность ложного срабатывания низкая. Системы киберобмана фиксируют действия против ложной цели – попытки подключения, команды, передаваемые файлы, методы авторизации – и передают эту информацию средствам мониторинга и реагирования. На основе собранных данных получают информацию о тактиках, инструментах и процедурах атакующего, благодаря чему можно оперативно локализовать и обезвредить угрозу.

Ложный FTP-сервер может быть применим для защиты файловых хранилищ, так как FTP – устаревший и простой протокол, который часто используется в уязвимых или тестовых средах. В классическом FTP команды и учетные данные передаются в открытом виде. Это делает такие серверы привлекательной целью при разведке и поиске слабой аутентификации.

Автоматические сканеры и боты активно ищут открытые FTP-службы и пытаются выполнить подбор паролей или загрузить файлы. Ложный FTP-сервер быстро фиксирует такие попытки, а также предоставляет ценную информацию о методах доступа, используемых командах и перемещениях атакующего внутри файловой структуры, без риска компрометации реальных данных.

Использование ложного FTP-сервера теоретически является действенным методом обнаружения атак. За счет сочетания низкой вероятности ложных срабатываний и привлекательности протокола для злоумышленников такая ловушка может служить ранним сигналом об атаке.

Тестирование ложного FTP-сервера

В качестве ложного FTP-сервера рассматривается FTP Honeypot [3]. Ловушка развернута на виртуальной машине в изолированной лабораторной сети. Служба прослушивала порт 2121. При необходимости возможен перенос порта на стандартный порт FTP 21. Источники подключений – локальная тестовая машина и вторая виртуальная машина в том же подсегменте. Ведение журналов происходило с записью временной метки, IP-адреса клиента, порта клиента и полученных FTP-команд. В ловушку не помещались реальные учетные данные и конфиденциальные файлы.

Заявленные характеристики [3]:

– ложная цель с низким уровнем взаимодействия: имитирует FTP-сервис для безопасного перехвата и регистрации попыток доступа;

– настраиваемая конфигурация сервера: аргументы командной строки позволяют легко настроить хост и порт;

– подробное ведение журнала событий: регистрирует каждую команду, отправленную на FTP-сервер, что помогает в анализе безопасности;

– взаимодействие в реальном времени: обеспечивает взаимодействие с клиентами в режиме реального времени, собирая учетные данные и команды;

– образовательный и исследовательский инструмент: идеально подходит для изучения моделей атак на основе FTP и для образовательных целей.

Установка требует наличия Python 3.x и библиотеки Twisted. Команды для установки [3]:

```
git clone https://github.com/0xNslabs/ftp-honeypot.git
cd ftp-honeypot
pip install twisted
```

Запуск производится следующей командой:

```
python3 sip.py --host 0.0.0.0 --port 2121
```

Как и заявлено, порт можно изменить, однако для использования стандартных портов необходимы права администратора.

Для тестирования использовалась установка напрямую в систему, однако для реальной эксплуатации целесообразнее использовать виртуальное окружение, например Python venv virtual environments.

Фрагмент журнала событий:

```
2025-10-05 15:05:06+0000 [-] SimpleFTPFactory starting on 2121
2025-10-05 15:06:55+0000 [__main__.SimpleFTPFactory] FTP NEW Connection Client IP: 127.0.0.1, Port: 35410
2025-10-05 15:07:01+0000 [SimpleFTPProtocol, 0, 127.0.0.1] Received data: b'USER user'
2025-10-05 15:07:08+0000 [SimpleFTPProtocol, 0, 127.0.0.1] Received data: b'PASS password'
2025-10-05 15:07:24+0000 [SimpleFTPProtocol, 0, 127.0.0.1] Received data: b'QUIT'
2025-10-05 15:07:53+0000 [_main_SimpleFTPFactory] FTP NEW Connection Client IP: 127.0.0.1, Port: 46374
2025-10-05 15:07:53+0000 [SimpleFTPProtocol, 1,127.0.0.1] Received data: b'USER test'
```

2025-10-05 15:07:53+0000 [SimpleFTPProtocol, 1,127.0.0.1] Received data: b'PASS 1234'

2025-10-05 15:07:53+0000 [SimpleFTPProtocol, 1,127.0.0.1] Received data: b'LIST'

2025-10-05 15:08:05+0000 [SimpleFTPProtocol, 1,127.0.0.1] Connection lost

2025-10-05 15:11:13+0000 [_main_SimpleFTPFactory] FTP NEW Connection Client IP: 192.168.186.129, Port: 58934

2025-10-05 15:11:28+0000 [SimpleFTPProtocol, 2, 192.168.186.129] Received data: b'USER user'

На основе этих данных получены следующие результаты:

– служба успешно запущена на порту 2121;

– зафиксированы тестовые подключения с разных адресов;

– в сессиях получены команды USER, PASS, LIST, QUIT;

– в журнале событий присутствуют введенные потенциальным злоумышленником данные;

– ловушка формировала поведение, позволяющее продолжить сеанс, имитируя реальный сервер (в частности – обработка команды LIST).

Заключение

В результате тестирования подтверждено, что развернутый FTP Honeypot надежно фиксирует попытки подключения, введенные учетные данные и базовые команды протокола. Журнал событий содержит достаточно информации для идентификации источников атак и воспроизведения последовательности действий. Это делает ловушку пригодной в качестве источника ранних сигналов о компрометации файловых ресурсов. Ложная цель показала способность имитировать поведение реального FTP-сервера достаточно правдоподобно, чтобы удерживать сессию и собрать дополнительные команды, что важно для накопления телеметрии, позволяющей отличать случайные сканирования от целенаправленной разведки.

Одновременно следует учитывать ограничение: низкоинтерактивная реализация не воспроизводит все аспекты полноценного сервера и может быстро привести к потере интереса потенциального злоумышленника.

Для практического внедрения можно порекомендовать следующие шаги:

– интегрировать журнал событий ловушки в центральную систему мониторинга событий и настроить корреляцию по IP, учетным данным и временным меткам;

– задать пороговые правила оповещений – повторные попытки авторизации и последовательность команд LIST/RETR/STORE должны поднимать приоритет тревоги;

– ограничить сетевое окружение ловушки и исходящий трафик, чтобы исключить ее использование атакующим в качестве плацдарма.

Благодаря тому, что все данные о подключениях и взаимодействии сохраняются в журнал событий в стандартном формате, возможно использование данного метода в составе общего комплекса мониторинга (SIEM) для обнаружения активности потенциальных злоумышленников наряду с другими подобными ложными целями, имитирующими различные информационные ресурсы, такие как базы данных [4].

В итоге, реализованный FTP Honeypot подтверждает теоретическую состоятельность подхода и может служить компонентом комплекса обнаружения компрометации файловых серверов при соблюдении мер безопасности и интеграции с существующими средствами реагирования.

Список литературы

1. 2024 Data Breach Investigations Report [Электронный ресурс]. – URL : https://www.verizon.com/business/resources/reports/2024-dbir-data-breach-investigations-report.pdf (дата обращения: 06.10.2025).

2. Why & How to Replace FTP [Электронный ресурс]. – URL : https://www.egnyte.com/guides/file-sharing/ftp-replacement (дата обращения: 07.10.2025).

3. Simple FTP Honeypot Server [Электронный ресурс]. – URL : https://github.com/0xNslabs/ftp-honeypot (дата обращения: 08.10.2025).

4. Применение технологии киберобмана для защиты баз данных / В. В. Алексеев, Д. А. Вислобоков, И. С. Шишкин, Е. С. Маленков // Актуальные проблемы кибербезопасности. Противодействие экстремизму и терроризму в информационной молодежной среде : сб. докл. I межрегион. науч.-практ. конф. – Брянск : БГТУ, 2025. – С. 16 – 21.

Кафедра «Информационные системы и защита информации»
ФГБОУ ВО «ТГТУ»