ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
«F6 Malware Detonation Platform»

Описание процессов, обеспечивающих поддержание жизненного цикла

Содержание

ТЕРМИНЫ И СОКРАЩЕНИЯ

Термин	**Определение**
АС	Автоматизированная Система
Заказчик	Зарегистрированный пользователь в системе заказчика передавший третьим лицам все необходимы данные и реквизиты для управления приложением или выполняющий указания третьих лиц за вознаграждение.
Исполнитель	Работы Исполнителя на протяжении всего жизненного цикла могут исполняться: •АО БУДУЩЕЕ; • Компанией-интегратором, по выбору Заказчика
ЛВС	Локальная вычислительная сеть
ОС	Операционная Система
ПО	Программное обеспечение F6 Malware Detonation Platform, MDP.
ТС	(«Технический Сервис») Система взаимодействия Заказчика, позволяющая обмениваться сообщениями и создавать цепочки обращений, которая представляет из себя отдельный раздел «Службу Поддержки» в панели управления «F6 Malware Detonation Platform». В случае недоступности указанных систем формат взаимодействия осуществляется через электронный почтовый ящик.

1 ОБЩИЕ СВЕДЕНИЯ

1.1 Введение

Настоящий документ содержит описание процессов поддержания жизненного цикла программного обеспечения «F6 Malware Detonation Platform» (далее – ПО, Malware Detonation Platform, MDP). Поддержание жизненного цикла ПО осуществляется за счет его сопровождения в течение всего периода эксплуатации и совершенствования (проведения обновлений) согласно собственному плану разработки и по заявкам Пользователей.

1.2 Назначение ПО

«F6 Malware Detonation Platform» представляет собой специализированное решение для анализа вредоносного программного обеспечения, которое запускает подозрительные файлы в изолированной среде (песочнице), чтобы безопасно изучить их поведение (поведенческий анализ). ПО позволяет детально отслеживать взаимодействие вредоносного ПО с операционной системой, реестром, файлами и сетевыми ресурсами, предоставляя полную картину его активности без риска заражения реальной информационной инфраструктуры. ПО не только анализирует простые вредоносные файлы, но и помогает выявлять эксплойты — программы, использующие уязвимости систем для выполнения атак. Платформа поддерживает различные типы файлов для анализа, включая исполняемые файлы, документы и скрипты.

1.3 Функциональные возможности ПО

ПО обладает следующими функциональными возможностями:

– Подсистема реализуется в виде программно-аппаратного комплекса для установки в стандартную 19-дюймовую стойку с наличием как минимум одного интерфейса 1000BASE-T для подключения к ЛВС и управления.

– Анализ потенциально вредоносных файлов осуществляется в изолированной среде для обнаружения ранее неизвестных угроз.

– Использование низкоуровневого монитора для выявления поведенческих маркеров.

– Применение элементов машинного обучения для анализа.

– Анализ файлов в изолированной среде проводится с использованием русифицированного образа ОС Windows.

– Архитектура подсистемы поддерживает гибкое горизонтальное масштабирование для повышения производительности.

– Подсистема поддерживает анализ файлов размером до 100 Мб.

– Возможность поведенческого анализа до 14,000 уникальных объектов в сутки.

– Подсистема поддерживает анализ файлов различных форматов (7z, exe, pdf и др.) в изолированной среде.

– Определение типов файлов по содержимому (Android application, архивы, Windows DLL и др.) для эффективного детектирования угроз.

– Поддержка ручной загрузки файлов на анализ с настройкой параметров виртуализации и анализа.

– Возможность подключения к виртуальной машине по протоколу RDP во время анализа файла.

– Поведенческий анализ архивов с паролем и поиск паролей в анализируемом контексте.

– Использование гипервизора для выполнения анализа.

– Статический сигнатурный анализ и противодействие техникам обхода систем поведенческого анализа (VMEvasion, Sandbox-evasion).

– Формирование отчета по результатам анализа с детализированной информацией о процессах, сетевой активности и файловой системе.

– Создание MITRE ATT&CK матриц с указанием задействованных тактик и техник.

– Запись экрана виртуальной машины за период проведения анализа.

– Извлечение конфигурационных файлов вредоносного ПО с указанием командных центров и параметров работы.

– Атрибуция проанализированных объектов к известным инструментам, группировкам и семействам вредоносного ПО.

– Экспорт анализируемых файлов и объектов для их дальнейшего анализа сторонними экспертами.

– Поддержка операционных систем Windows XP, Windows 7, Windows 10 (x64/x86) и Android 7.

– Поддержка локализации на русском и английском языках.

– Возможность изменения маршрута для вывода трафика из виртуальных машин.

– Кастомизация виртуальных машин, включая изменение домена, имени компьютера и пользователя.

– Автоматический подбор конфигурации виртуальной машины для анализа экземпляров вредоносного ПО.

– Эмуляция пользовательской активности (движения мыши, нажатие клавиш, переключение между окнами).

– Использование технологии CV.

– Извлечение и запуск дополнительных команд из реестра.

– Поддержка поддельного SMTP-сервера для проведения анализа.

– Включение Mitmproxy для анализа.

2 Процесс разработки ПО

Процесс разработки ПО включает в себя:

- Сбор и анализ требований к разработке ПО;
- Проектирование архитектуры ПО;
- Разработка ПО в коде;
- Проведение тестирования ПО перед эксплуатацией;
- Запуск в промышленную эксплуатацию ПО;
- Промышленная эксплуатация ПО;
- Сопровождение ПО.

2.1 Сбор и анализ требований к разработке ПО

На этапе сбора и анализа требований ПО определяются требования всех заинтересованных сторон, включая функциональные и нефункциональные требования.

Основные этапы сбора и анализа требований к разработке ПО:

- Определение основных задач и целей, которые должен решить проект ПО;
- Определение ключевых заинтересованных сторон (заказчики, пользователи, разработчики, другой персонал);
- Сбор требований к ПО;
- Анализ требований, их уточнение, пересмотр на точность и реализуемость;
- Оценка рисков;
- Создание плана и графика реализации проекта;
- Документирование требований и проектных планов;
- Согласование и утверждение требований.

2.2 Проектирование архитектуры ПО

Проектирование архитектуры ПО – это процесс определения общей структуры системы, ее компонентов и модулей, а также взаимодействия между компонентами системы на основе выработанных требований.

Проектирование архитектуры включает в себя следующие этапы:

– Определение архитектурного стиля;

– Определение основных модулей и компонентов системы, их взаимодействие;

– Выбор технологий (языки программирования, базы данных и т.д.) и инструментов для разработки ПО;

– Документирование архитектуры системы.

2.3 Разработка ПО в коде

На этапе разработки ПО в коде осуществляется реализация проектных решений с помощью выбранных технологий и инструментов.

Разработка ПО включает в себя следующие этапы:

– Написание исходного кода ПО с использованием выбранных технологий и инструментов;

– Проверка кода на наличие ошибок и несоответствий;

– Проведение интеграционного тестирования;

– Отладка кода (исправление обнаруженных ошибок);

– Проверка кода для улучшения качества ПО, его производительности и безопасности;

– Интеграция частей кода и модулей ПО в единую систему, проверка их совместимости;

– Подготовка к тестированию ПО перед эксплуатацией.

2.4 Проведение тестирования ПО перед эксплуатацией

Тестирование ПО перед эксплуатацией – это оценка качества ПО, его функциональности, производительности и безопасности. Цель тестирования заключается в подтверждении того, что ПО удовлетворяет установленным требованиям и корректно работает в различных условиях. Тестирование включает в себя следующие этапы:

– Автоматические unit-тесты, встроенные в pipeline CI/CD процессов при работе с репозиторием хранения исходного кода;

– Автотесты с использованием Allure (pytest + selenium) с покрытием не менее 35%;

– Полуавтоматическое регрессионное тестирование на каждую вновь добавляемую новую функцию;

– Ручное тестирование: интеграционное, функции, API.

После ручного тестирования и добавления каждой функции в ПО, ручное тестирование автоматизируется и добавляется в общий банк автотестов.

2.5 Запуск ПО в промышленную эксплуатацию

Запуск в промышленную эксплуатацию – это процесс подготовки окружения для развертывания ПО на целевой среде Заказчика, а также установка и активация модуля. Запуск в промышленную эксплуатацию осуществляется силами Исполнителя.

Для корректного проведения данного тестирования, необходимо выполнение пунктов по оценке работоспособности «F6 XDR».

ПО разворачивается в сетевой инфраструктуре заказчика.

2.5.1 Локальное размещение Malware Detonation Platform

2.5.1.1 Проверка физической работоспособности MDP

1. Проверить наличие и размещение оборудования Системы.

Результат: Сервер установлен в серверную стойку.

2. Проверить подачу питания на серверы Системы.

Результат: наличие подключения блоков питания сервера к сети электропитания.

3. Проверить интеграцию с инфраструктурой заказчика.

Результат: необходимые сетевые интерфейсы подключены к локальной сети заказчика.

4. Убедиться, что оборудование включается при нажатии кнопки включения.

2.5.1.2 Проверка корректности загрузки исполняемого программного обеспечения MDP

1. После корректной загрузки программного обеспечения на устройствах Системы отображается поле для ввода логина и пароля на вход в программную оболочку.

2. После входа в программную оболочку проверить статус соединения с MXDR Console: Huntbox connection - OK.

2.5.2 Облачное размещение Malware Detonation Platform

2.5.2.1 Проверка доступности модуля

Перейти в раздел **Настройки** → **Модули**, должен присутствовать модуль “Cloud Malware Detonation”, индикатор модуля с левой стороны должен быть зеленым.

2.5.2.2 Проверка работоспособности локально размещенного модуля MDP и проведение анализа файла

Данный пункт распространяется только на инсталляции с локально размещенным MXDR Console.

1. Проверить, корректность настроек вывода виртуальных машин MDP в интернет:

– При наличии физического сетевого доступа MDP в Интернет (а также при наличии доступа к общедоступным DNS, например, 8.8.8.8) в разделе **Настройки** → **Модули** → **Malware Detonation Platform** → **Основные настройки** → **Доступ виртуальных машин в Интернет** → Требуется, Маршрут для вывода трафика → через mgmt-порт;

– При отсутствии физического сетевого доступа MDP в Интернет, но наличии доступа в Интернет у MXDR Console (а также при наличии доступа к общедоступным DNS, например, 8.8.8.8) в разделе **Настройки** → **Модули** → **Malware Detonation Platform** → **Основные настройки** → **Доступ виртуальных машин в Интернет** → Требуется, Маршрут для вывода трафика → через MXDR Console.

– При отсутствии физического сетевого доступа MDP и MXDR Console в Интернет, в разделе **Настройки** → **Модули** → **Malware Detonation Platform** → **Основные настройки** → **Доступ виртуальных машин в Интернет** → Отключен.

2. Проверить, что индикатор состояния устройства (слева) в разделе **Настройки** → **Модули** → **MDP** зеленый.

3. Перейти в раздел **Расследование** → **Проверенные файлы** → **Загрузить файл** → **Загрузить**. Если доступ в Интернет у ВМ MDP отсутствует, в сайдбаре загрузки файла настройку “Соединение с интернетом” перевести в состояние Выключено.

4. В разделе **Расследование** → **Проверенные файлы** установить фильтр по источнику файла в значение **Manual**. Наличие записи о завершенном анализе файла, загруженного на предыдущем этапе,и наличие отчета MDP свидетельствуют о корректности работы модуля. Предельное время ожидания результата анализа - 60 минут.

2.5.2.3 Проверка работоспособности модуля MDP (облачное размещение) и проведение анализа файла

Данный пункт распространяется и на инсталляции с облачной MXDR Console, и на инсталляции с локальной MXDR Console.

1. Проверить, что индикатор состояния устройства (слева) в разделе **Настройки** → **Модули** → **Malware Detonation Platform** зеленый.

2. Перейти в раздел **Расследование** → **Проверенные файлы** → **Загрузить файл** → **Загрузить**.

3. В разделе **Расследование** → **Проверенные файлы** установить фильтр по источнику файла в значение **Manual**. Наличие записи о завершенном анализе файла, загруженного на предыдущем этапе, и наличие отчета MDP свидетельствуют о корректности работы модуля. Предельное время ожидания результата анализа - 60 минут.

2.6 Промышленная эксплуатация

Промышленная эксплуатация (далее – Эксплуатация) – это этап жизненного цикла, когда установленное ПО используется в реальных рабочих условиях на постоянной основе.

Эксплуатация включает в себя следующие этапы:

- Аналитическое сопровождение и работы по выявлению аномалий и мошеннической активности среди клиентов Заказчика;
- Обработка выявляемых событий и предоставление обратной связи;
- Тонкая настройка правил выявления мошеннической активности;
- Контроль работоспособности АС Заказчика;
- Контроль работоспособности ПО;
- Доработка и регулярное обновление ПО для устранения ошибок, повышения производительности, а также введения новых функций;
- Периодическая отчетность по работоспособности и устранениям неисправностей ПО;
- Поддержка актуальной документации.

2.7 Сопровождение ПО

В течение всего периода эксплуатации ПО Заказчику предоставляется сопровождение ПО, в рамках которого оказываются следующие услуги:

- Техническая поддержка Пользователей;
- Решение инцидентов (экстренных неисправностей), возникающих в процессе эксплуатации ПО;
- Устранение сбоев и ошибок, выявленных в ПО;
- Совершенствование ПО;
- Мониторинг производительности ПО;
- Оптимизация эффективности работы ПО;
- Поддержка актуальной технической документации по ПО;
- Уведомление об обновлениях и изменениях ПО;
- Обучение новых пользователей.

3 Совершенствование ПО

ПО на постоянной основе подвергается развитию и улучшению в рамках процессов:

– развития и добавления новых функциональных возможностей, позволяющих расширить области применения ПО;

– оптимизации работы модулей ПО, обеспечивающей повышение производительности, скорости обработки данных и отказоустойчивости;

– обновления пользовательского интерфейса.

Совершенствование ПО происходит за счет проведения модернизаций ПО в соответствии с собственным планом доработок, а также с учетом заявок клиентов по вопросам испытаний установки и эксплуатации, полученных через ТС.

4 Устранение неисправностей ПО

Неисправности, которые были выявлены в ходе полноценной эксплуатации ПО, могут быть исправлены следующими способами:

- Массовое обновление компонентов ПО;
- Единичная работа технического специалиста по запросу Пользователя.

В случае возникновения неисправности клиент направляет заявку через Технический Сервис (далее – ТС) Разработчика с подробным описанием воспроизведенной проблемы (версия ПО, описание конфигурации, версия приложения клиента, прикрепленные скриншоты). Затем технический специалист проводит следующие действия:

- подтверждает наличие неисправности в соответствии с описанием проблемы от Заказчика;
- тестирует неисправность в функционале ПО и создает отчет по результатам тестирования;
- фиксирует задачу на исправление проблемы в текущий или ближайший релиз обновления ПО или консультирует клиента по корректности выполняемых действий.

Задачи по устранению неисправностей в функционале ПО осуществляются силами Разработчика. В соответствии с внутренним планом выхода обновлений подсистемы предоставляется исправленный механизм работы ПО.

Процессы по устранению неисправностей протекают непрерывно, без остановки функционирования ПО.

4.1 Устранение экстренных неисправностей ПО

В экстренном случае, когда ошибка препятствует полноценному использованию функционала ПО, группа разработчиков готовит внеплановый выход обновления или предоставляет исправленную версию ПО.

При возникновении экстренных неисправностей Заказчик отправляет запрос через ТС со следующими данными:

- Четко сформулированная тема обращения;
- Версия приложения заказчика, на которой осуществляется эксплуатация ПО;
- Версия ПО;
- Пошаговое описание воспроизведения ошибки;

- Скриншоты, демонстрирующие наличие найденной ошибки.

5 Техническая поддержка

Техническая поддержка Пользователей осуществляется в соответствии с условиями контракта следующими способами:

- Приоритетный способ осуществления техподдержки через создание запросов во вкладке «Поддержка» по ссылке https://xdr.f6.security/service-desk
- по электронной почте: info@f6.ru;
- по номеру телефона: +7 495 984-33-64;

В рамках технической поддержки оказываются следующие услуги:

- консультация по фактическому наличию имеющегося функционала в системе;
- помощь в настройке и интеграции ПО;
- помощь в эксплуатации ПО;
- решение технических проблем;
- пояснение принципов работы имеющихся механизмов ПО;
- поиск, тестирование и фиксирование найденных ошибок;
- предоставление актуальной документации по настройке, эксплуатации и работе ПО.

Время работы технической поддержки: с понедельника по пятницу с 9:00 до 18:00 UTC+3.

Фактический адрес размещения службы поддержки ПО «F6 Malware Detonation Platform»: 115088, г. Москва, ул. Шарикоподшипниковская, д. 1

6 Информация о персонале

6.1 Персонал, обеспечивающий работу ПО на рабочих местах пользователей

К эксплуатации ПО допускаются лица, ознакомившиеся с документацией по эксплуатации ПО в разделе «Помощь» пользовательского интерфейса ПО.

К эксплуатации ПО привлекается штатный персонал Заказчика, имеющий следующие навыки:

- навыки работы с персональным компьютером на уровне опытного пользователя;
- опыт работы с электронными документами;
- опыт использования web-браузеров;
- Знания в соответствующей предметной области.

6.2 Персонал, обеспечивающий техническую поддержку, аналитическую поддержку и модернизацию ПО «F6 Malware Detonation Platform»

Специалисты, обеспечивающие техническую и аналитическую поддержку и развитие ПО, должны обладать следующими знаниями и навыками:

- знание функциональных возможностей ПО;
- знание особенностей работы с ПО;
- знание языков программирования, исходя из должностных обязанностей: Python, GO, Rust, JavaScript, TypeScript;
- знание реляционных и не реляционных БД, исходя из должностных обязанностей: PostgreSQL, Elasticserch, ClickHouse, MongoDB;
- знание средств мониторинга производительности серверов.

Должность	Компетенции	Выполняемые работы	Количество специалистов
Системный программист-разработчик	C, C++, Rust, Python	Техническая поддержка; Аналитическое сопровождение;	5

		Разработка и совершенствование ПО.	
Анатилик разработки детектирующей логики	Python, Rust, Sigma	Техническая поддержка; Аналитическое сопровождение; Разработка и совершенствование ПО.	5
DevOps инженер	Linux, Ansible, Docker, GitLab CI\CD, Elasticsearch, PostgreSQL, Minio.	Техническая поддержка; Аналитическое сопровождение; Совершенствование ПО.	2
Тестировщики	Allure, Pytest, Gitlab CI/CD, Разработка авто тестов, функционального и нагрузочного тестирования	Разработка тест-планов для тестирований продуктового функционала; Проведение ручного тестирования согласно разработанным планам; Разработка автоматических тестов и анализ его результатов; Проведение регрессионного тестирования.	2

ИНФОРМАЦИЯ О ФАКТИЧЕСКИХ АДРЕСАХ

Фактический адрес размещения разработчиков ПО 115088, г. Москва, ул. Шарикоподшипниковская, д. 1

Фактический адрес размещения службы поддержки ПО 115088, г. Москва, ул. Шарикоподшипниковская, д. 1

Контакты службы поддержки:

- Электронная почта: info@f6.ru
- Телефон: +7 495 984-33-64

Информация о фактическом адресе размещения инфраструктуры разработки ПО

ПО может поставляться в виде облачного сервиса, в таком случае ПО размещается на удаленных серверах компании АО «Селектел» по адресу:

188683, Ленинградская область, Всеволожский район, г.п. Дубровка, ул. Советская, дом 1, литера Б.