OTUS

Специальная цена

Реверс-инжиниринг. Продвинутый курс

Длительность курса: 88 академических часов

1 Анализ программ

1	**Написание плагинов под Ida**	разбирается с IDA SDK и IDA python для написания расширений, позволяющих упростить анализ бинарного кода Домашние задания 1 String Decryptor Цель: Цель: научиться писать плагины Результат: получить готовый к использованию плагин Реализация плагина расшифровки строк в трояне TinyNuke. На вход плагину подаётся указатель на данные, их размер и ключ, а на выходе выдаётся расшифрованная строка
2	**Использование динамической бинарной инструментации (Pin, Angr)**	разбираем PIN фреймворк и реализовываем свои PIN toolы на примере хука функций и обнаружения утечек памяти
3	**Кастомизация и сборка виртуальных машин скрытия от ВПО (Pafish)**	кастомизация виртуальных машин требуется для максимального сокрытия среды виртуализации от вредоносного ПО. Как правило, это особенно нужно для автоматических анализаторов, к примеру , sandbox, но также нужен и вирусным аналитикам.
4	**Техники антиэмуляции**	рассматриваем различные техники, используемые вредоносными программами, которые используются в попытке

		обойти методы эвристического детектирования
5	**Настройка Cuckoo Sandbox**	настраиваем систему автоматического анализа ВПО
6	**Малварные скрипты (возможности ps1, vbs/vbe, js/jse)**	изучаем возможности скриптовых языков, которыми пользуется ВПО, а также некоторые приёмы деобфускации
7	**Использование легитимных утилит вредоносным ПО**	разбираем как и зачем ВПО используют легитимный софт для своих целей

2 Техники внедрение кода

1	**Техники инжектов (SetWindowsHook, openprocess/Virtual Allocate/CreateThread, AppInitDll)**	разбор техники на практике
2	**Process hollowing**	
3	**User Mode Apc Inject**	
4	**Atom Bombing**	
5	**Doppelganging**	
6	**Reflective DLL Injection**	

Домашние задания

1 Инжектор

Цель: Цель: Разобраться в инжектах путём написание программы
Результат: Навык “выцепливания” инжектов

Нужно реализовать внедрение любого кода в АП любого процесса, используя одну из выученных техник

3 Rootkit-ы

1	**DKOM**	разбор руткита, скрывающего процессы и файлы
2	**Driver-Filter of file system**	пример руткита для подмены содержимого определённого файла
3	**Kernel Mode APC inject**	разбор APC инжектов
4	**WFP драйвера**	разбор сетевой технологии WFP Домашние задания 1 WFP фильтр Цель: Цель: разобраться в фильтрации Результат: драйвер Необходимо реализовать драйвер-фильтр, который будет заменять в HTTP протоколе указанную подстроку, на определённую фразу
5	**SSDT hook (x86)**	пример осуществления подмены системных вызовов в ядре OS Домашние задания 1 DUMP SSDT Цель: Цель: изучить таблицу системных вызовов Результат: программа дампа таблицы Необходимо вывести таблицу из названия системных вызовов, их номеров и адресов
6	**mbr rootkit**	разбор буткита
7	**Архитектура защиты в Windows 10**	разбор новых методов защиты в OS Windows 10, основанных на базе виртуализации

4 Trojan-Ransom

1	**CryptoApi**	разбор криптографических библиотек, на примере реализации алгоритмов шифрования AES и RSA. Использование этих алгоритмов на примере разбора шифровальщиков
2	**CryptoPP**	

5 Техники эксплуатации

1	**Heap spray**	техника эксплуатации при UAF уязвимости на практике
2	**Stack Pivoting**	техника переключение стека
3	**Rop**	способы построения ROP гаджетов Домашние задания 1 ROP цепочка Цель: Цель: Понять принцип техники обхода DEP Результат: ROP шеллкод Необходимо построить пример ROP цепочки и выполнить произвольный код
4	**Jit-Spray**	разбор техники
5	**Разбор HEVD (Integer overflow)**	разбор уязвимостей в драйвере и их эксплуатация Домашние задания 1 Разработка эксплойта к тестовому драйверу 1 Цель: Цель: разобрать уязвимость Результат: эксплойт Необходимо проэксплуатировать уязвимость в тестовом драйвере и повысит привилегии процессу калькулятора
6	**Разбор HEVD (Stack overflow)**	
7	**Разбор HEVD (Type Confusion)**	
8	**Разбор HEVD (Pool overflow)**	Домашние задания 1 Разработка эксплойта к тестовому драйверу 2 Цель: Цель: разобрать уязвимость Результат: эксплойт Необходимо проэксплуатировать уязвимость в тестовом драйвере и повысит привилегии процессу калькулятора