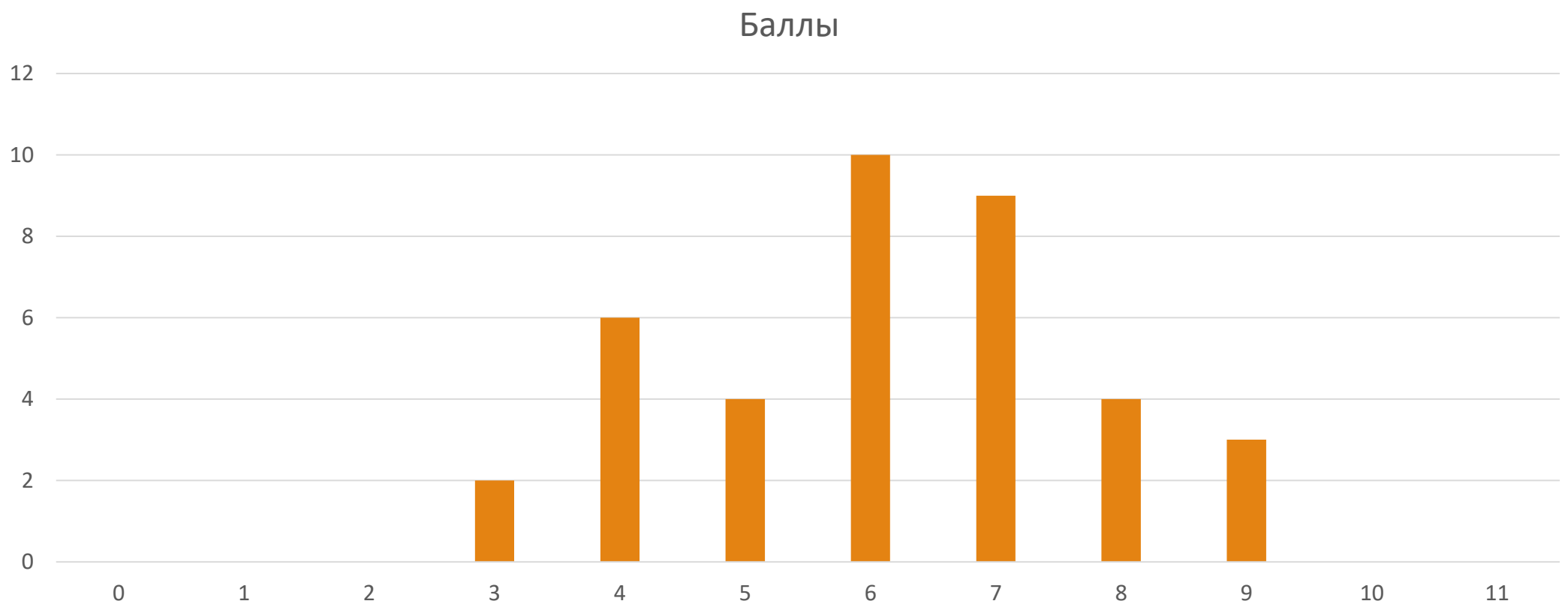
ОБРПО. Лекция 10 Привилегии

ТЮРИН КАЙ АНДРЕЕВИЧ

Но сначала результаты теста



ROP-цепочки

ROP цепочка позволяет составить исполнимый код из уже существующего в памяти (то есть нет необходимости передавать исполнимый код в чистом виде) – обход DEP.

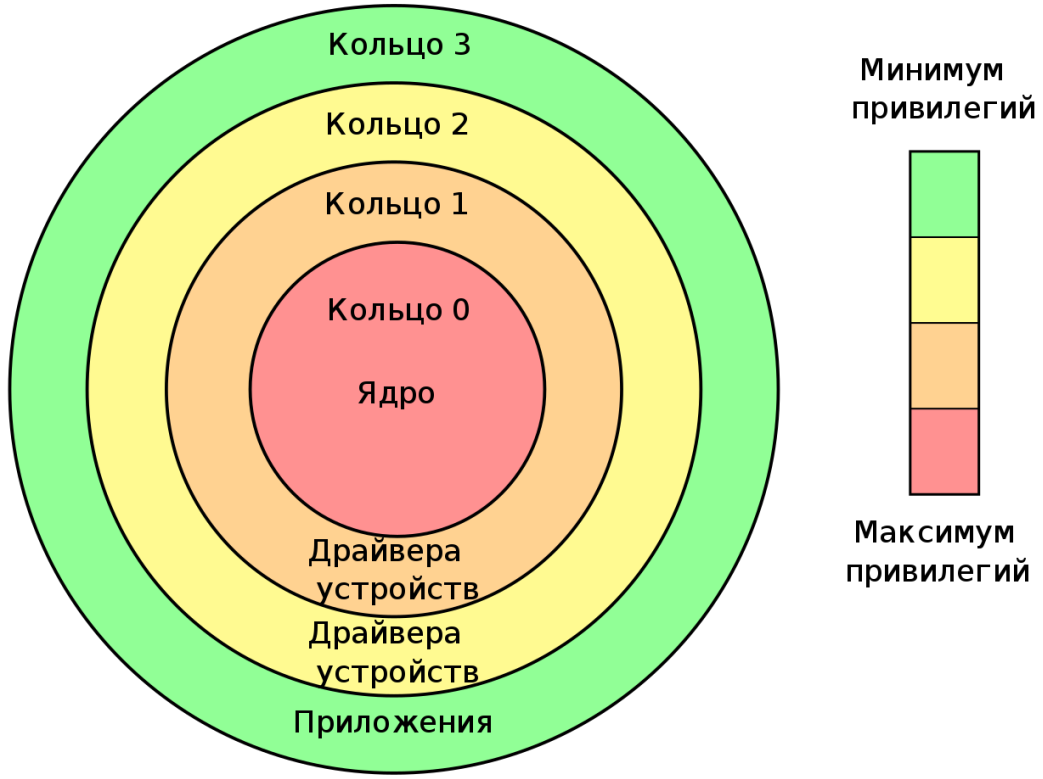
При этом необходимо знать расположение существующего кода в памяти (нет обхода ASLR)

Также для исполнения ROP-цепочки необходимо обойти stack canarie.

Кольца привилегий Windows

Современные процессоры архитектуры x86/x64 предоставляют 4 кольца защиты.

Однако Windows и большинство Unix-систем используют только два из них: нулевое и третье



nmap

nmap позволяет:

- сканировать порты

- определять программное обеспечение

- определять версию ПО

Для эксплуатации уязвимостей используются другие утилиты (например, mimikatz)

TLS

TLS предоставляет шифрование данных, аутентификацию клиента и сервера (зачитывалось даже если аутентификация клиента не отмечена).

Эксплуатация XSS вообще никак не зависит от (не)использования TLS.

HSTS

HSTS заставляет браузер придерживаться политики «подключение только по HTTPS». Для этого используются разные механизмы, например, невозможность перейти на HTTP страницу или соединиться при ошибках сертификата.

HSTS никак не влияет на CSRF.

HSTS в каком-то смысле может способствовать DoS атакам.

Инъекции

Инъекции возможны в любой язык. Для этого достаточно, чтобы в строку на данном языке каким-то (плохим) образом попадали данные от пользователя.

Авторизация

Авторизация – предоставление прав и их проверка.

Аутентификация – проверка подлинности (аутентичности) чего-то.

Выдача уникального дескриптора – идентификация.

Защита от SQL-инъекций

Защититься от SQL-инъекций можно экранированием входных параметров, а также посредством использования параметризованных запросов.

Второй вариант на самом деле не всегда возможен.

Фильтрация данных на предмет наличия ключевых слов/спецсимволов является порочной практикой.

Использование нестандартных имен таблиц и полей лишь немного увеличивает время для злоумышленника на атаку.

А теперь про привилегии

Привилегии в Unix

Не самый лучший пример, но:

1. достаточно показательный;

2. с распределением прав в Unix-подобных системах так или иначе придётся иметь дело.

Что есть субъект и объект?

Субъектом всякого действия является процесс

Объектами являются файлы, директории, сокеты, другие процессы, память, дескрипторы файлов, другие устройства

Чем определяются права процесса?

User ID: 32-разрядное целое число

Group ID: 32-разрядное целое число

Операции над файловой системой

Действия над файлами: чтение, запись, исполнение, смена атрибутов…

Действия над директориями: открытие, создание, удаление, переименование…

В Unix-системах типов прав на объекты файловой системы всего три: чтение, запись, исполнение.

Эти права имеют разные ограничения для владельца файла, группы-владельца и всех остальных.

Атрибуты файла

Пользователь-владелец, группа-владелец и матрица прав:

	R	W	X
Owner	1	1	0
Group	1	0	0
Other	0	0	0

Также биты объединяют по три и обозначают десятичным числом. (640 в данном случае)

Execute для директории – lookup.

То есть

Когда процесс делает вызов open(“/etc/passwd”)

Система проверяет у него права

x на /

x на /etc

x на /etc/passwd

Задача

У нас есть группы:

«третий курс ФИИТ» - fiit3

«старосты» - headmen

Как сделать файл, доступный на чтение только старостам с третьего курса ФИИТ без создания новых групп?

Решение

/foo/bar/grades

Где /foo имеет владельцем fiit3 с разрешённым “x”

/foo/bar имеет владельцем headmen с разрешённым “x”

File Descriptor

Проверка происходит только при получении дескриптора, а не при обращении по нему.

Процесс, имеющий доступ к файлу, может открыть его и передать дескриптор другому процессу.

Такой механизм может использоваться как для усиления безопасности, так и для её ослабления.

Процессы

Что можно сделать с процессом?

отладка (ptrace), посылка сигналов, ожидание выхода, получение статуса и т.д.

Для отладки и посылки сигналов нужно иметь тот же UID (это почти так, на практике есть много различных вариантов того, как это устроено).

Ожидание/получение статуса: необходимо быть родительским процессом.

Память

Один процесс не может обращаться к памяти другого процесса.

Исключение составляют механизмы отладки и отображение файлов в память.

Сеть

Какие есть сетевые операции?

1. слушать порт – ограничение на порты меньше 1024 (доступны только для UID 0)

2. соединяться с некоторым адресом – можно любому процессу

3. читать/писать – можно для любого процесса, обладающего дескриптором

4. посылать принимать raw пакеты – только для root’а

На уровне сети работают проверки сетевого экрана

Как устанавливаются атрибуты?

Системные вызовы для установки атрибутов доступа процессу:

setuid, setgid, setgroups

Их может вызывать только root (UID 0) (ну, почти)

Например, программа login запускается от root’а, после чего запускает шел от нужного пользователя по введённым данным

/etc/shadow – для проверки пароля

/etc/passwd – для сопоставления имени пользователя и UID

/etc/group – для проверки принадлежности пользователя группам

Setuid (suid) binaries

Исполнимые файлы, которые исполняются от имени своего владельца, а не пользователя, их запускающего.

Примеры:

/bin/su

/bin/sudo

Смена владельца файла

Производится при помощи команд

chown/chgrp

Доступны только суперпользователю (на нижнем уровне это один и тот же системный вызов)

Защита от эксплуатации suid-бинарей

Команда chroot – доступна только для root’а

Интуитивно она меняет понимание того, что такое /

На самом деле:

1. Меняет / на /foo

2. Не позволяет выйти на уровень выше, чем /

Как обойти ограничение на выход?

```
chroot(“/foo”)
foo = open(“/”)
chroot(“/bar”)
fchdir(foo)
chdir(“..”)
```

Ограничение на chroot

Чтобы избежать вышеописанной ситуации выхода за пределы корневой системы.

Чтобы suid-бинари не путались с тем, что такое /etc/passwd и т.д.

Итого

Система прав Unix даёт нам гибкую систему распределения прав, но для её настройки необходим суперпользователь

Права в Windows

Атрибуты доступа

Объекты имеют дескриптор безопасности SD (security descriptor), содержащий следующую информацию:

- идентификатор безопасности (SID) владельца объекта;
- идентификатор безопасности первичной группы владельца;
- дискреционный список контроля доступа (discretionary access control list — DACL);
- системный список контроля доступа (system access control list - SACL).

Список SACL управляется администратором системы (для аудита). Список DACL предназначен для идентификации пользователей и групп, которым предоставлен или запрещен определенный тип доступа к объекту. Этот список редактируется владельцем объекта, но и члены группы администраторов по умолчанию имеют право на смену разрешений на доступ к любому объекту, которое может быть у них отнято владельцем объекта.

Пример SID

S-1-5-21-1463437245-1224812800-863842198-1128

SID	Группа	Использование
S-1-0-0	Nobody (Никто)	Используется, когда SID неизвестен
S-1-1-0	Everyone (Все)	Группа, включающая всех пользователей за исключением анонимных
S-1-2-0	Local (Локальная)	Пользователи, вошедшие в терминалы, которые локально (физически) подключены к системе
S-1-3-0	Creator Owner ID (ID владельца создателя)	Идентификатор безопасности, который будет заменен идентификатором безопасности того пользователя, который создал новый объект. Этот SID используется в наследуемых ACE-элементах
S-1-3-1	Creator Owner ID (ID владельца создателя)	Идентификатор безопасности, который будет заменен идентификатором безопасности основной группы того пользователя, который создал новый объект. Этот SID используется в наследуемых ACE-элементах
S-1-5-18	Учетная запись Local System	Используется службами
S-1-5-19	Учетная запись Local System	Используется службами
S-1-5-20	Учетная запись Network Service	Используется службами

Списки прав

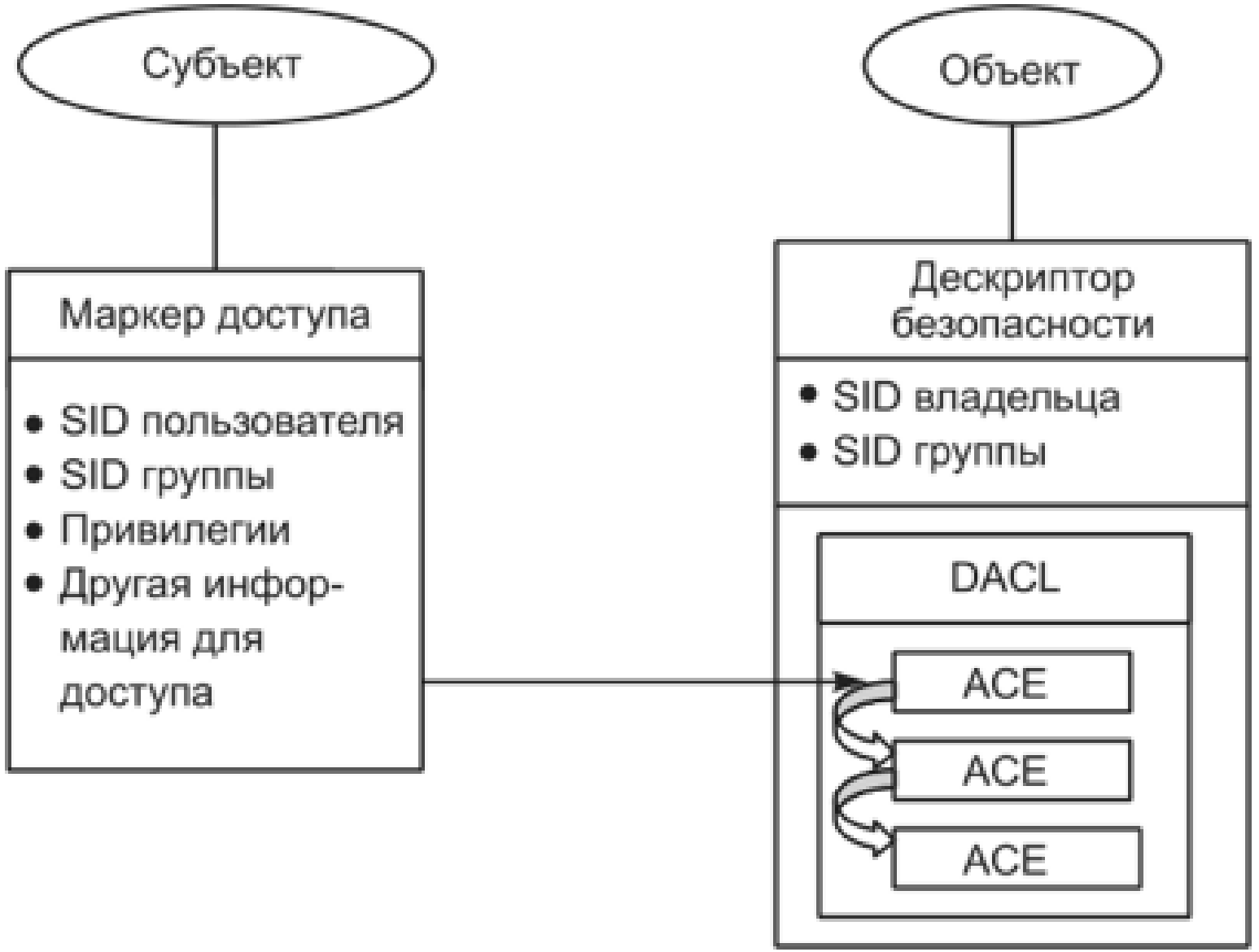
Каждый элемент списка DACL (access control entry — ACE) определяет права доступа к объекту одного пользователя или группы. Каждый ACE содержит следующую информацию:

• идентификатор безопасности SID субъекта, для которого определяются права доступа;

• маску доступа (access mask — AM), которая специфицирует контролируемые данным ACE права доступа;

• тип ACE;

• признак наследования прав доступа к объекту, определенных для родительского объекта.


Субъект
Объект
Маркер доступа
• SID пользователя
• SID группы
• Привилегии
• Другая инфор-
мация для
доступа
Дескриптор
безопасности
• SID владельца
• SID группы
DACL
ACE
ACE
ACE

Типы ACE

Элементы списка DACL могут быть двух типов — элементы, разрешающие специфицированные в них права доступа (Access-allowed ACE), и элементы, запрещающие определенные в них права доступа (Access-denied ACE). Элементы для запрещения субъектам использования определенных прав доступа должны размещаться в «голове» списка, до первого из элементов, разрешающих использование субъектом тех или иных прав доступа.

Типы доступа

Право доступа субъекта к объекту означает возможность обращения субъекта к объекту с помощью определенного метода (типа) доступа. В операционной системе Windows различаются специальные, стандартные (общие) и родовые (generic) права доступа к объектам. Специальные права доступа определяют возможность обращения к объекту по свойственному только данной категории объектов методу — чтение данных из объекта, запись данных в объект, чтение атрибутов объекта, выполнение программного файла и т. д. Стандартные права доступа определяют возможность доступа к объекту по методу, применимому к любому объекту, — изменение владельца объекта, изменение списка DACL объекта, удаление объекта и т. д.

Типы прав для файлов и папок

• полный доступ (включает в себя все специальные и стандартные разрешения);

• изменение (все разрешения, кроме «Удаление подпапок и файлов», «Смена разрешений» и «Смена владельца»);

• чтение и выполнение (включает разрешения на «Обзор папок (выполнение файлов)», «Содержание папки (чтение данных)», «Чтение атрибутов», «Чтение дополнительных атрибутов», «Чтение разрешений», «Синхронизация»);

• список содержимого папки (только для папок); включает в себя те же разрешения, что и «Чтение и выполнение», но они наследуются по-разному;

• чтение (включает в себя право на «Содержание папки (чтение данных)», «Чтение атрибутов», «Чтение дополнительных атрибутов», «Чтение разрешений», «Синхронизацию»);

• запись (включает в себя разрешения «Создание файлов (запись данных)», «Создание папок (дозапись данных)», «Запись атрибутов», «Запись дополнительных атрибутов», «Чтение разрешений», «Синхронизацию»).

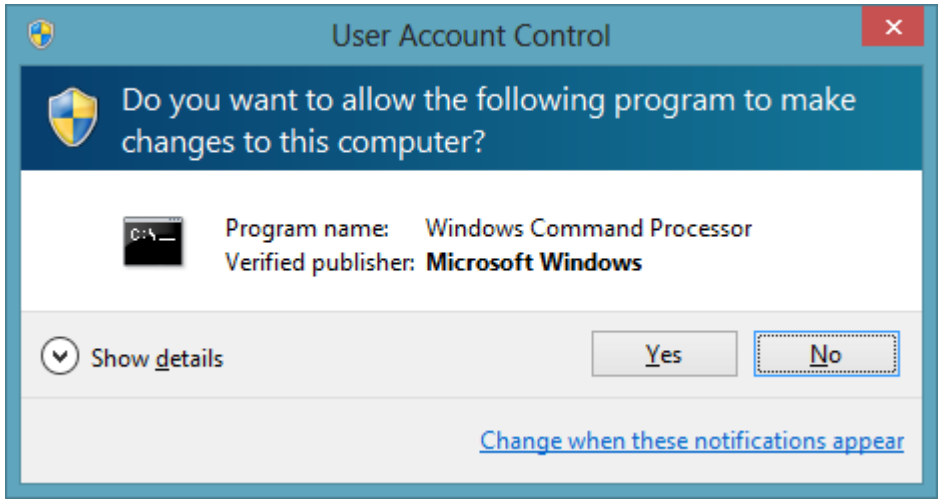
UAC

UAC – User Account Control

Табличка, которая просит у администратора подтверждения его действий

“UAC is not a security feature”

Используется исключительно для «защиты от дурака», потому что пользователь и так является администратором и имеет все необходимые права



Итого

Распределение прав в Windows более сложное в своей основе, но более гибкое и удобное для конечного пользователя.