

# **CTF на Физтехе**

Занятие 1

# Компьютерная безопасность

- <https://github.com/xairy/mipt-ctf#План-на-2015-2016>
- Зачем?
  - Общая осведомленность (пользователь vs программист)
  - Bug bounty
  - Fun
- Heartbleed (2011 - 2014), Shellshock (1989 - 2014)
- Adobe Flash exploits от Hacking Team (2015)
- White hat и black hat

# CTF

- CTF == Capture The Flag == Захват Флага
- `flag{73c0487d1b4c9326bc4ec5ac09bf69eb}`
- Attack-Defense - у каждой команды есть сервер с несколькими уязвимыми сервисами. Для получения очков нужно находить уязвимости в этих сервисах, использовать их для атаки сервисов команд противников и исправлять уязвимости у своих сервисов.
- Jeopardy - множество заданий на разные тематики. За решение заданий команды получают очки. Сложнее задание - больше очков.
- <https://ctftime.org/> - расписание CTF соревнований

# Типы заданий

- Exploitation
- Reverse Engineering
- Cryptography
- Forensics
- Web
- Networking
- Reconnaissance
- ...

# Программа

<https://github.com/xairy/mipt-ctf#План-на-2015-2016>

# Формат занятий

- Лекция + практика
- Лекция: презентация + скринкаст
- Практика: workshop или самостоятельно

# СВЯЗЬ

- Репозиторий курса: <https://github.com/xairy/mipt-ctf>
- Группа ВК: [https://vk.com/mipt\\_ctf](https://vk.com/mipt_ctf)
- Почтовая рассылка: [mipt-ctf@googlegroups.com](mailto:mipt-ctf@googlegroups.com)

# УК РФ

- [Статья 146. Нарушение авторских и смежных прав](#)
- [Статья 272. Неправомерный доступ к компьютерной информации](#)
- [Статья 273. Создание, использование и распространение вредоносных программ для ЭВМ](#)
- [Статья 274. Нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети](#)



**Вопросы?**

# Bash

- Bourne Again SHell - командный процессор
- Однострочные команды или простые скрипты: `bash`
- Сложные скрипты: Python, Go, ...

# Навигация

- `ls [dir] [-a] [-l]` Вывести содержимое директории
- `cd dir` Перейти в директорию dir
- `cd ..` Перейти в родительскую директорию
- `cd ~` Перейти в домашнюю директорию
- `pwd` Распечатать путь текущей директории

# Создание и удаление

- `touch file` Создать пустой файл `file`
- `rm file` Удалить файл `file`
- `mkdir dir` Создать новую пустую директорию `dir`
- `rm -r dir` Удалить директорию `dir` и ее содержимое

# Копирование и перемещение

- `cp file path` Скопировать файл
- `mv file path` Переместить или переименовать файл
- `cp -r dir path` Скопировать директорию
- `mv dir path` Переместить или переименовать директорию

# Содержимое файлов

- `cat file` Вывести содержимое файла
- `less file` Отобразить содержимое файла с плюшками
- `head -n 2 file` Вывести первые две строки файла
- `tail -n 2 file` Вывести последние две строки файла

# Перенаправление ввода и вывода

- `cmd > file` Записать вывод команды в файл
- `cmd >> file` Приписать вывод команды к файлу
- `cmd < file` Подать содержимое файла на ввод команде
- `cmd1 | cmd2` Подать вывод одной команды на ввод другой

# Поиск по содержимому файлов

```
grep [opts] "pattern" file
```

```
cat file | grep [opts] "pattern"
```

**-v**

Вывести строки, где не встречается образец

**-a**

Искать по бинарным файлам

**-i**

Игнорировать регистр символов

**-E**

Расширенные регулярные выражения

**-n**

Печатать номера строк

**-o**

Вывести только совпадающую с образцом часть строки



# Регулярные выражения

```
egrep [opts] "pattern" file
```

```
cat file | egrep [opts] "pattern"
```

**^**    **"^license"**    Начало строки

**(abc)**    **"(license)"**    Группа символов

**\$**    **"license\$"**    Конец строки

**.**    **"Li.en.e"**    Любой символ

**[abc]**    **"[0-9]"**    Любой символ из перечисленных

**\***    **"\[a-z]\*\"**    Повторить ноль или больше раз

**[^abc]**    **"[^ ]1"**    Любой кроме перечисленных

**+**    **"\[a.+\\"**    Повторить один или больше раз

**[ab]{8}**    **"w{3}"**    Повторяется несколько раз

**?**    **"(copy)?right"**    Повторить ноль или один раз

# Обработка текста

- `cat file | cut -c 2-5` Вывести символы со 2 по 5 каждой строки
- `cat file | sort` Отсортировать строки по алфавиту
- `cat file | uniq` Удалить одинаковые строки, идущие подряд
- `cat file | sort | uniq` Вывести уникальные строки

# Обработка текста

```
awk 'program' file
```

```
cat file | awk 'program'
```

```
'{print $0}'
```

Вывести каждую строку

```
'{print $1}'
```

Вывести первое слово каждой строки

```
'{print "1: " $1 ", 2: " $2}'
```

Вывести для каждой строки:  
1: <слово 1>, 2: <слово 2>

# Поиск файлов

```
find path [opts]
```

```
find . | grep [opts] pattern
```

```
-name "*.txt"
```

Искать по имени файла

```
-type f
```

Искать только файлы

```
-type d
```

Искать только директории

# Работа с архивами

- `zip out.zip path` Сжать файл или папку с помощью zip
- `unzip file.zip` Распаковать zip архив
- `tar -czf out.tar.gz path` Сжать с помощью tar и gzip
- `tar -xzf file.tar.gz` Распаковать tar.gz архив

# Netcat

- `man nc` Работа с TCP/UDP соединениями
- `nc -l 9999` Ожидать соединения по TCP на порту 9999
- `nc 127.0.0.1 9999` Присоединиться к 9999 порту на localhost

# More

- `vim file` Редактор файлов (альтернативы: emacs, nano, ...)
- `lynx site.com` Браузер
- `file, hexdump, sed, strings, curl, wget, ssh, scp, ...`
- <https://github.com/jlevy/the-art-of-command-line>

# Документация

- `man cmd` Документация по команде `cmd`
- <https://www.google.com/>



# Практика

<https://github.com/xairy/mipt-ctf/tree/master/01-intro/01-bash>

**Вопросы?**

**Спасибо за внимание!**