# Анализ кода и информационная безопасность

Лекция 02

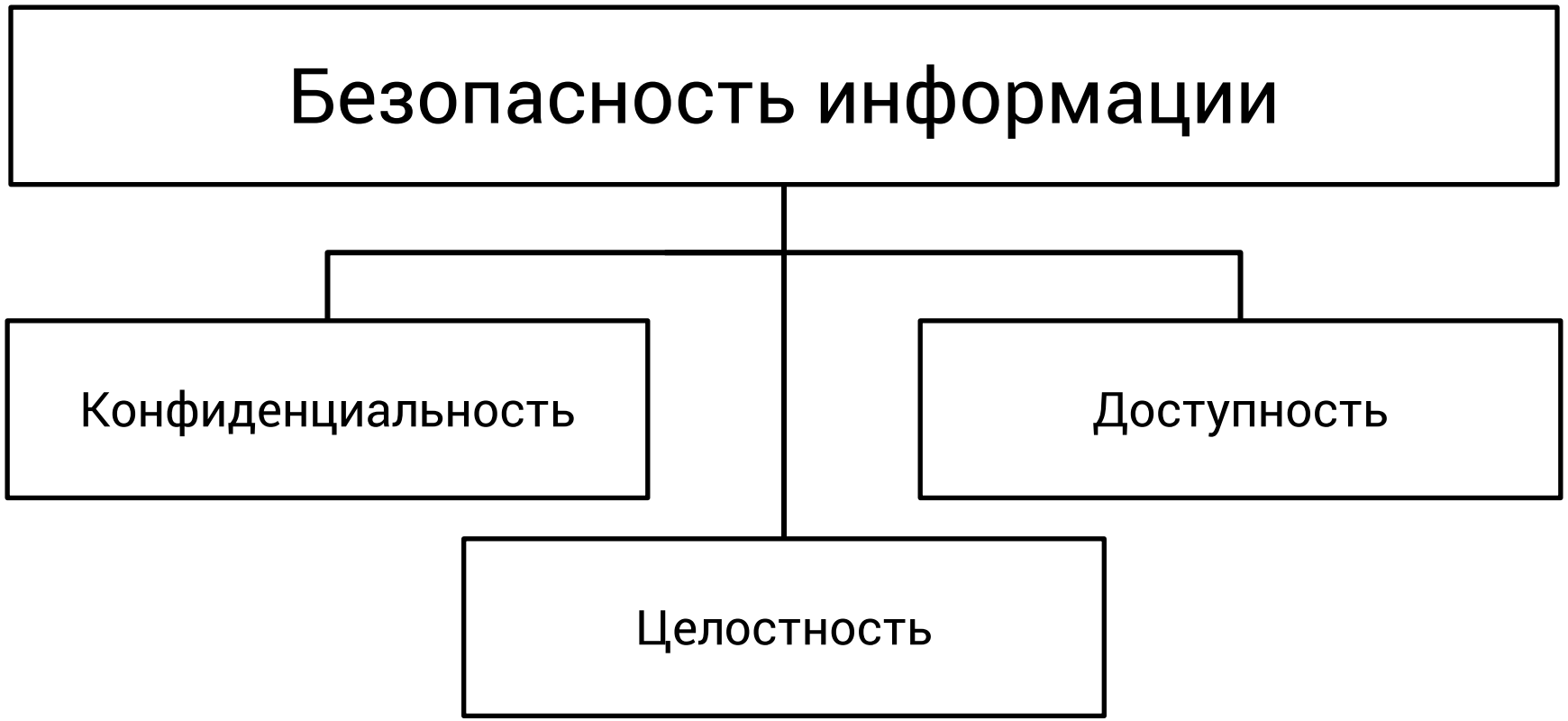
Основные понятия безопасности информации

МГУ / ВМК / СП

0201

# ОСНОВНЫЕ ПОНЯТИЯ БЕЗОПАСНОСТИ ИНФОРМАЦИИ

# Безопасность информации



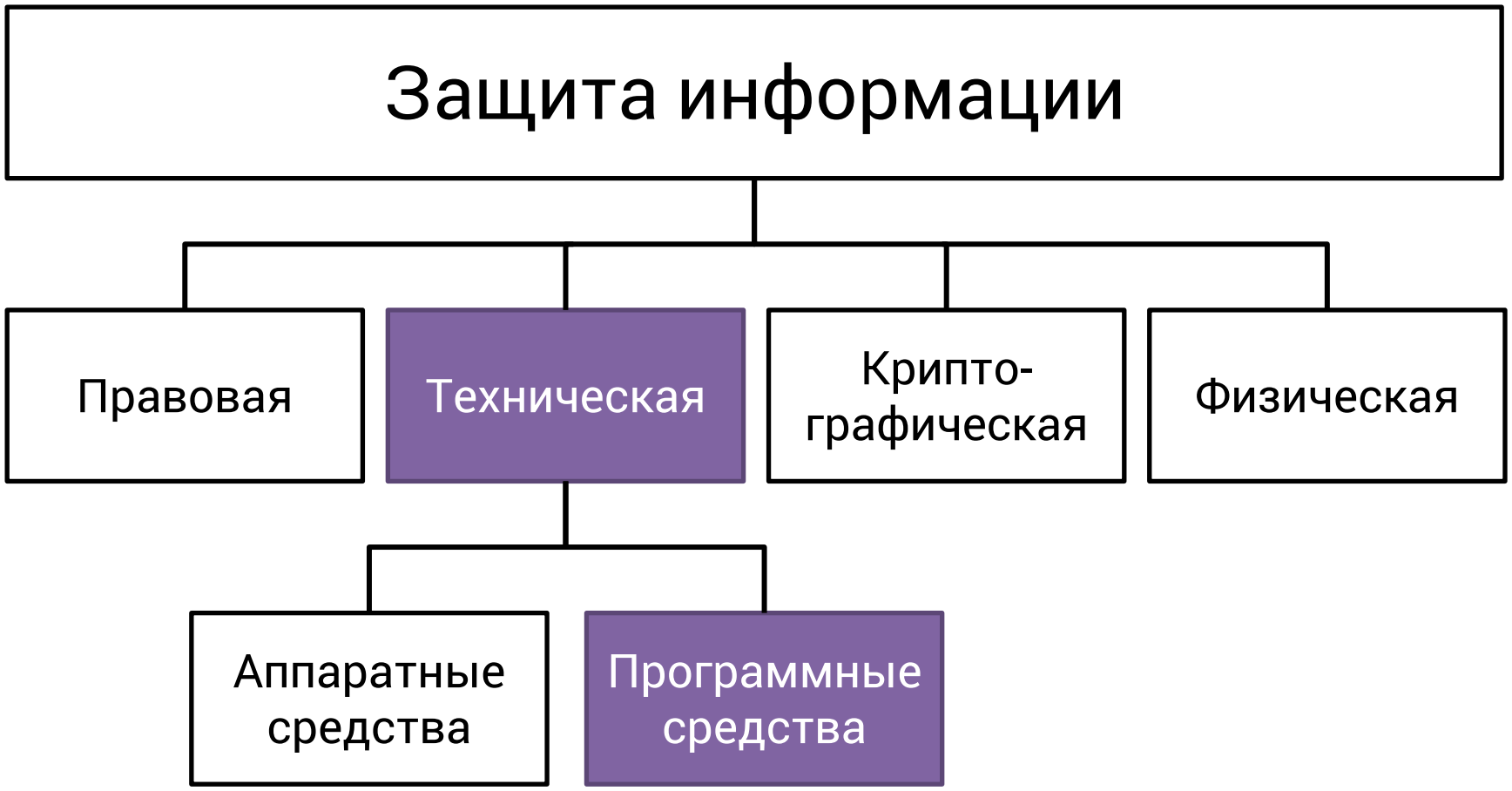
# Безопасность информации

Конфиденциальность информации — состояние информации, при котором доступ к ней осуществляют только субъекты, имеющие на него право.

Целостность информации — состояние информации, при котором отсутствует любое её изменение либо изменение осуществляется только преднамеренно субъектами, имеющими на него право.

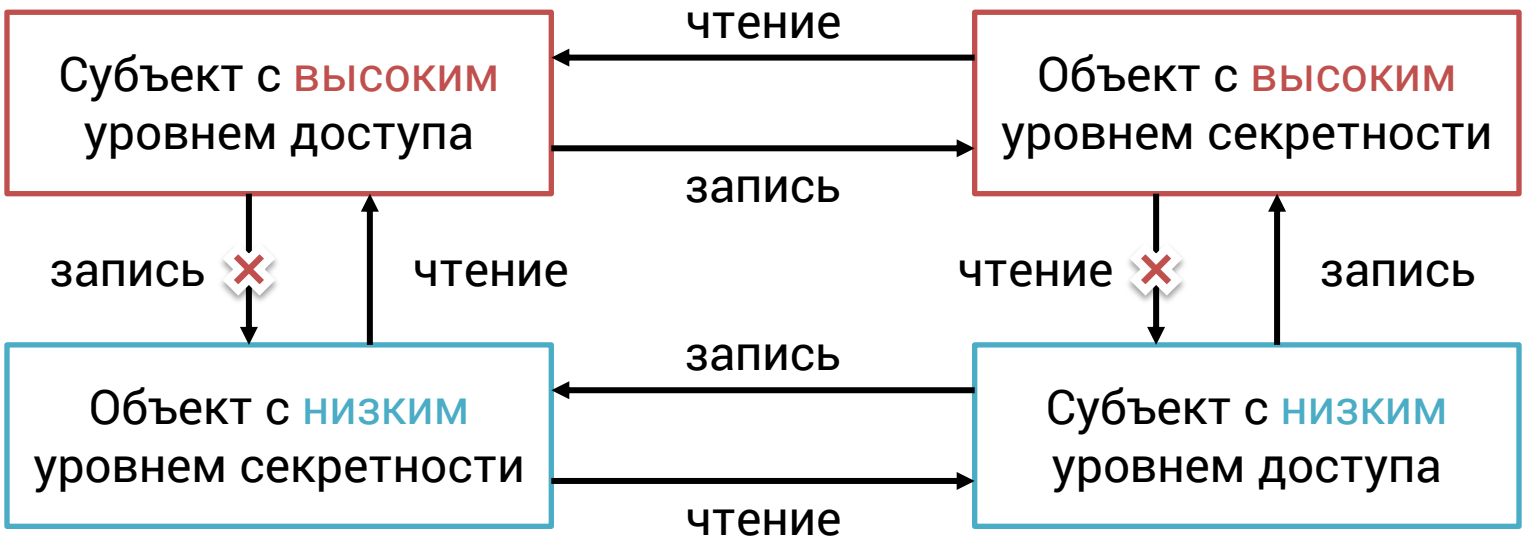
Доступность информации — состояние информации, при котором субъекты, имеющие права доступа, могут реализовать их беспрепятственно.

# Защита информации



# Политики безопасности

Политика безопасности — набор правил, в соответствии с которыми производится обработка информации.




Модель Белла–Лападулы

# Безопасность информационных систем

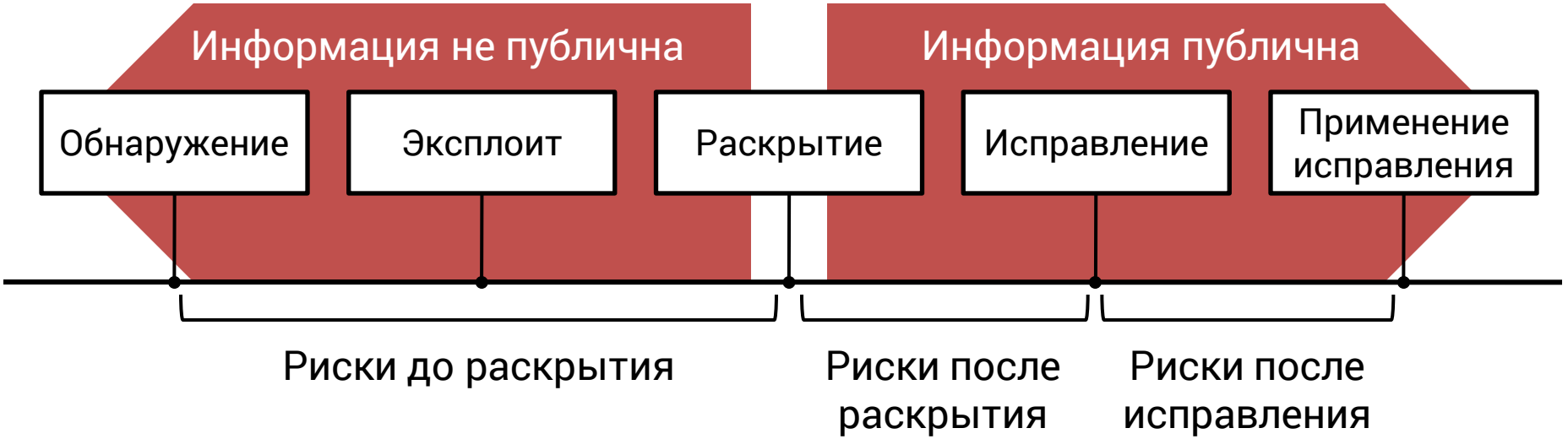
Уязвимость информационной системы — свойство информационной системы, обусловливающее возможность реализации угроз безопасности обрабатываемой в ней информации.

Угроза безопасности информации — совокупность условий и факторов, создающих потенциальную или реально существующую опасность нарушения безопасности информации.

# Жизненный цикл уязвимости

Окно уязвимости — время от возникновения уязвимости до исправления соответствующей ошибки или обеспечения невозможности атаки.

Уязвимость нулевого дня (zero-day) — уязвимость, для которой на данный момент нет исправления или способа защиты.



# Эксплоит

Эксплоит — программа, фрагмент кода или последовательность команд, использующие уязвимость, чтобы добиться непредусмотренного поведения информационной системы.

В идеале атакующий пытается перехватить поток управления уязвимой системы, чтобы выполнить произвольный код:

- полезная нагрузка — внедряемый код;
- шелл-код — полезная нагрузка, дающая доступ к интерпретатору команд ОС.

# Ошибки в программах

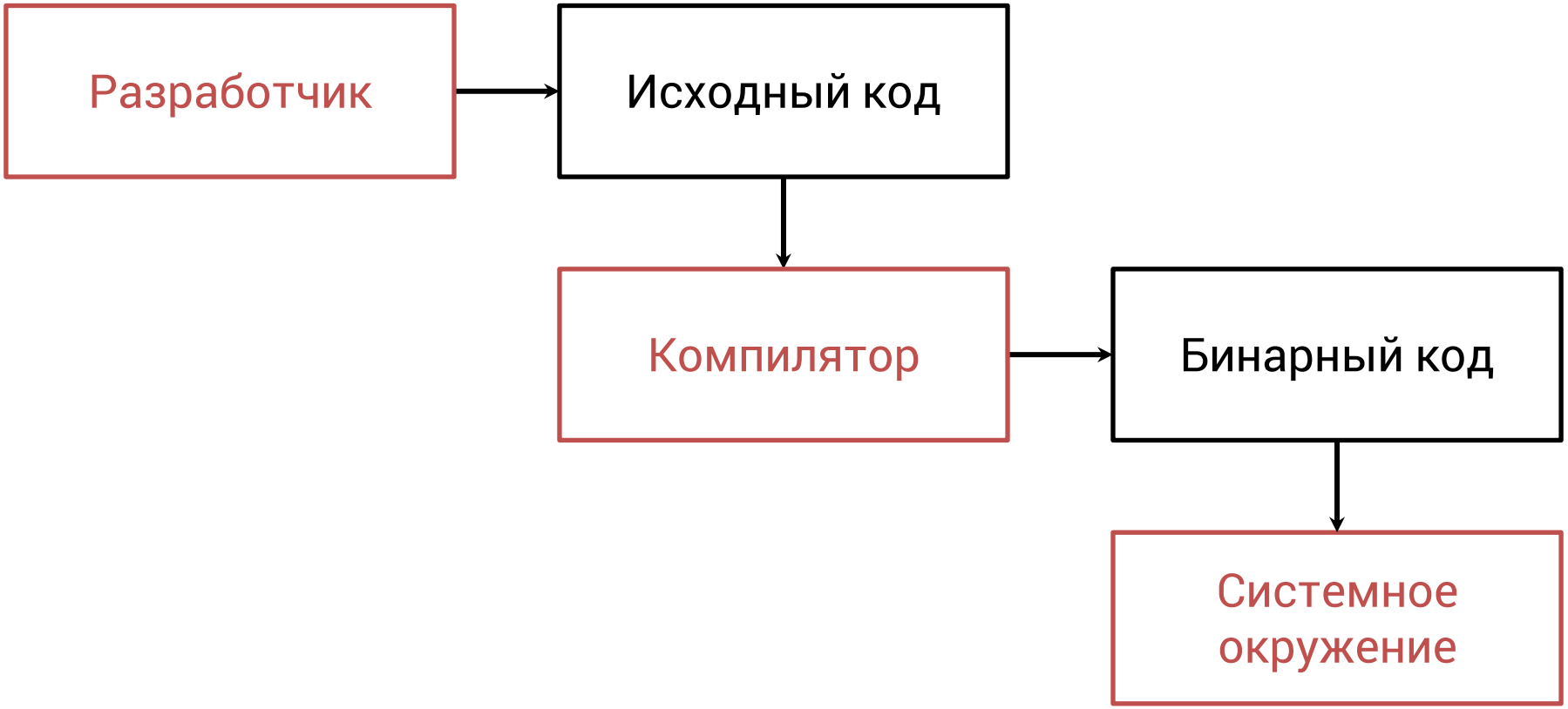
**Ошибки в логике приложения**

**Ошибки уровня архитектуры приложения**
(неверная работа с разделяемыми ресурсами в многопоточном ПО)

**Несоответствие модели языка программирования**
(выход за границы массива, нарушение правил алиасинга, ABI)

**Несоответствие требованиям со стороны аппаратуры**
(неверная адресация в памяти, целочисленное деление на 0)

# Ошибки и закладки



# Нарушения безопасности в ПО

1. Нарушение **конфиденциальности**:
   - OpenSSL Heartbleed;
   - программная закладка в роутере;
   - шпионаж через веб-камеру в телевизоре.
2. Нарушение **целостности**:
   - программная закладка в роутере;
   - изменение базы данных с паролями через внедрение SQL-кода.
3. Нарушение **доступности**:
   - аварийное завершение программы;
   - отказ в обслуживании — DoS, DDoS.

# Последствия ошибок

1. Перехват потока управления:
   - возможность нарушения всех трёх аспектов безопасности информации;
   - может быть осуществлён скрытно.
2. Ошибки времени выполнения:
   - нарушение доступности, а в некоторых случаях и целостности;
   - потенциально при редком стечении обстоятельств возможно также и нарушение конфиденциальности;
   - как правило, достаточно быстро обнаруживаются;
   - сравнительно легче поддаются автоматическому выявлению.
3. Безопасность информационных потоков.

0202

# КЛАССИФИКАЦИЯ ОШИБОК

# Классификация ошибок

1. Классификация по ГОСТ Р 50546–2015:
   - порядка 20 типов недостатков, приводящих к уязвимостям, простое перечисление.
2. MITRE Common Weakness Enumeration (CWE):
   - обширная база знаний, систематизация типов ошибок и взаимосвязей между ними.
3. HP Enterprise Security Fortify Taxonomy:
   - древовидная организация;
   - группировка по языкам и «царствам».
4. Банк данных угроз безопасности информации ФСТЭК:
   - плоский список, порядка 200 типов угроз;
   - источники, объект воздействия, последствия.

# MITRE CWE

- Сайт: https://cwe.mitre.org/.
- Иерархическая модель — от общего к частному.
- Этап внесения ошибки (проектирование, реализация…).
- Языки программирования.
- Возможные угрозы конфиденциальности, целостности, доступности.
- Связи между разными типами ошибок — «предшествует», «следует за», «также является».
- Примеры: фрагменты кода и конкретные уязвимости в распространённом ПО.
- Предотвращение и противодействие.

# CWE-126: “Buffer Over-read”

- Improper Access of Indexable Resource (‘Range Error’)
  Неверный доступ к индексируемому ресурсу (‘Ошибка диапазона’)
  - Improper Restriction of Operations within the Bounds of a Memory Buffer
    Неверное ограничение операций внутри границ буфера памяти
    - Access of Memory Location After End of Buffer
      Доступ к участку памяти после конца буфера
      - Buffer Over-read
        Чтение за пределами буфера

# CWE-126: “Buffer Over-read”

Описание: the software reads from a buffer using buffer access mechanisms such as indexes or pointers that reference memory locations after the targeted buffer.

Время внесения ошибки: реализация ПО.

Языки программирования: C, C++.

Угроза конфиденциальности: чтение памяти.

Следует за: Improper Null Termination.

Пример: OpenSSL Heartbleed.

# OpenSSL Heartbleed

# OpenSSL Heartbleed

Сервер, пошли мне обратно эти **135** байтов: **TEST**

Клиент

**TEST**)4)лz+УФымq=/†EcЌMуЊ?b6e°a e_ещзХњР‡ЖуУ‘ьТЫЉ+ЬЃІЫн‰aЋВ 1ClDN†WKыb c\a™vZAbШ‹ЎМ>ЖГа Ќ ?%сфПБS? ¢9Ї°РлYґжsЎ"Цск‘Б’фю _2rjђВ

Сервер

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

https://cwe.mitre.org/documents/vulnerability_theory/intro.html

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 1 — CWE-120: “Classic Buffer Overflow”

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 2 — CWE-23: “Relative Path Traversal”

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 3 — CWE-79: “Failure to Preserve Web Page Structure (XSS)”

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 4 — CWE-134: “Uncontrolled Format String”

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 5 — CWE-476: “NULL Pointer Dereference”

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 6 — CWE-20: “Improper Input Validation”

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 7 — CWE-116: “Improper Encoding or Escaping of Output”

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 8 — CWE-73: “External Control of File Name or Path”

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 9 — CWE-404: “Improper Resource Shutdown or Release”

# “Bug barrel”

```
 1  printf("<title>Blissfully Ignorant, Inc.</title>");
 2  ftype = Get_Query_Param("MessageType");
 3  strcpy(fname, "/home/cwe/");
 4  strcat(fname, ftype);
 5  strcat(fname, ".dat");
 6  handle = fopen(fname, "r");
 7  while (fgets(line, 512, handle)) {
 8      if (strncmp(line, "<script>", 8))
 9          printf(line);
10  }
11  return 200;
```

Ошибка № 10 — CWE-252: “Unchecked Return Value”

# Оценка критичности ошибки

Common Vulnerability Scoring System (CVSS) — способ численного выражения степени критичности ошибки (сайт — https://www.first.org/cvss).

Группы метрик:

- сложность реализации уязвимости;
- степень нарушения конфиденциальности, целостности и доступности;
- взаимосвязь между уязвимым компонентом системы и тем, безопасность которого нарушается.

# CVSS

## Метрики сложности реализации уязвимости

1. **AV** — вектор атаки:
   - **N** — атака по сети;
   - **A** — атака по локальной сети;
   - **L** — атака в рамках одной машины;
   - **P** — атака с физическим доступом.
2. **AC** — сложность воспроизведения:
   - **L** — низкая;
   - **H** — высокая.
3. **PR** — требуемые полномочия:
   - **N** — отсутствуют;
   - **L** — низкие;
   - **H** — высокие.
4. **UI** — взаимодействие с пользователем:
   - **N** — не требуется;
   - **R** — требуется.

# CVSS

## Метрики степени нарушения безопасности

1. **C** — степень нарушения конфиденциальности информации:
   - **H** — высокая;
   - **L** — низкая;
   - **N** — нарушения конфиденциальности нет.
2. **I** — степень нарушения целостности информации:
   - **H** — высокая;
   - **L** — низкая;
   - **N** — нарушения целостности нет.
3. **A** — степень нарушения доступности информации:
   - **H** — высокая;
   - **L** — низкая;
   - **N** — нарушения доступности нет.

# CVSS

## Метрика S (Scope)

**S** — изменение контекста авторизации:

- **U** — отсутствует;
- **C** — присутствует.

Критерий S:C — уязвимый компонент и компонент, в котором нарушается безопасность информации, различны.

Пример: ошибка в реализации ПО виртуальной машины позволяет произвести действия в хостовой ОС изнутри гостевой ОС.

# CVSS

## OpenSSL Heartbleed

AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

7.5 из 10

1. Ошибка тем критичнее, чем легче эксплуатируется уязвимость, которую она порождает.
2. Ошибка тем критичнее, чем в большей степени нарушаются конфиденциальность, целостность и доступность системы.
3. Изменение контекста авторизации существенно увеличивает критичность ошибки.

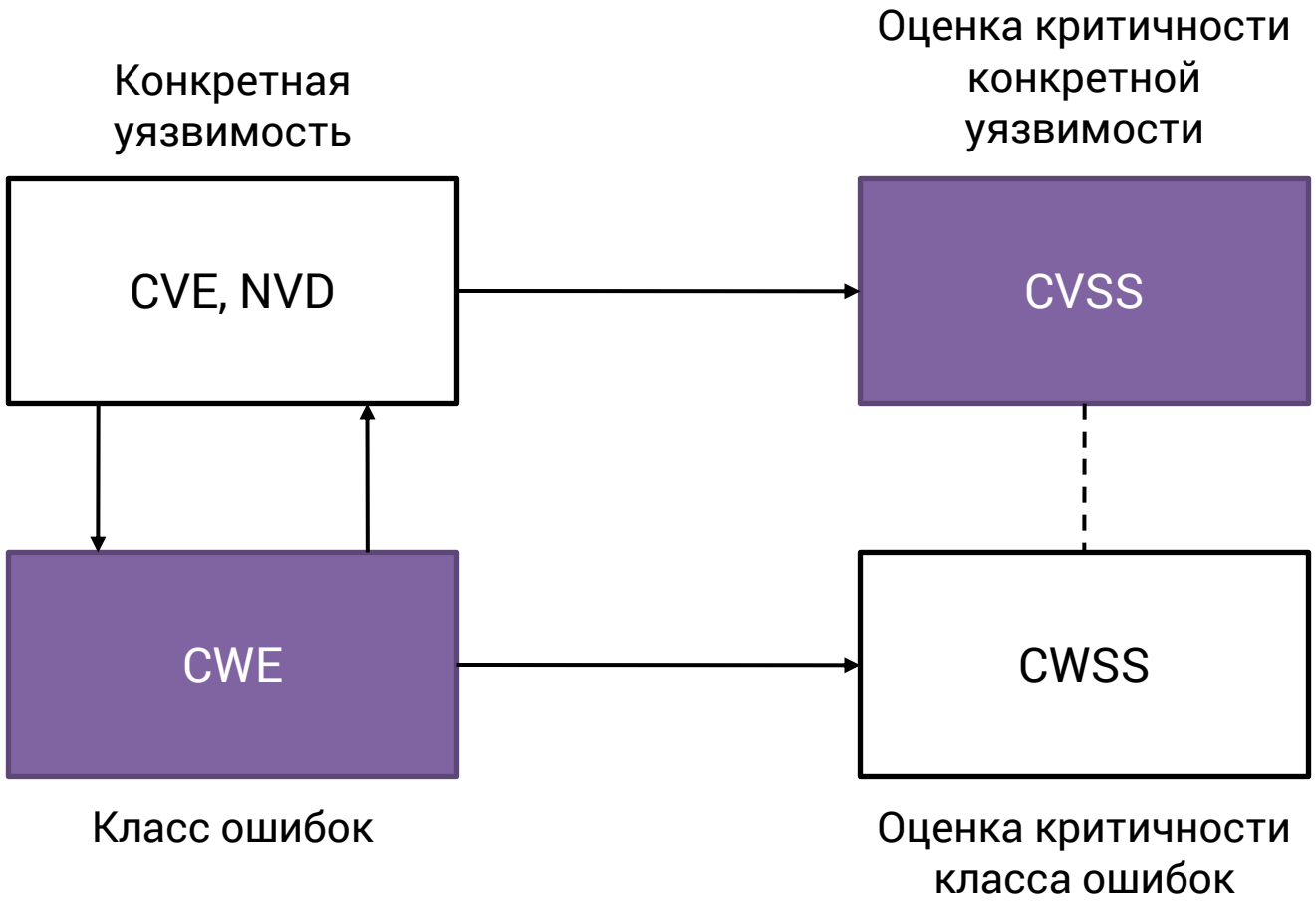
# Уязвимости и эксплоиты

## Информационные ресурсы

1. Бюллетени разработчиков и исследователей.
2. Реестры:
   - MITRE Common Vulnerabilities and Exposures (CVE) — https://cve.mitre.org/;
   - National Vulnerability Database (NVD) — https://nvd.nist.gov/;
   - Банк данных угроз безопасности информации ФСТЭК (БДУ) — http://www.bdu.fstec.ru/vul.
3. Эксплоиты, шелл-код:
   - Exploit Database — https://www.exploit-db.com/;
   - Metasploit — https://www.metasploit.com/.

# Уязвимости и эксплоиты

## Взаимосвязь информационных ресурсов



# Литература к лекции

## Основные источники


1. Brian Chess, Jacob West. Secure Programming with Static Analysis / Addison-Wesley Professional, 2007:
   - глава 1 — “The Software Security Problem.”
2. ГОСТ Р 50922–2006. Защита информации. Основные термины и определения.
3. ГОСТ Р 50.1.056–2005. Техническая защита информации. Основные требования и определения.
4. ГОСТ Р 56546–2015. Защита информации. Уязвимости информационных систем. Классификация уязвимостей информационных систем.
5. The Heartbleed Bug.
6. MITRE Common Weakness Enumeration.
7. Common Vulnerability Scoring System.
8. MITRE Common Vulnerabilities and Exposures.

# Литература к лекции

## Дополнительные источники

1. MITRE Common Weakness Scoring System.
2. Банк данных угроз безопасности информации.